



FACULTAD DE INFORMÁTICA

TESINA DE LICENCIATURA

Título: Visualización de información en la web dirigida por usuarios finales

Autora: Josefina Estevez de Urquiza

Directores: Diego Torres, Sergio Firmenich

Codirector: –

Asesor profesional: –

Carrera: Licenciatura en Sistemas

Resumen

En este trabajo se implementa una herramienta de visualización para ser utilizada en aplicaciones web previamente existentes. Uno de los objetivos es que sea utilizada en aplicaciones dentro del marco de ciencia ciudadana. De esta manera pretende democratizar la información y simplificar el trabajo a la hora de mejorar interfaces, reduciendo la barrera tecnológica entre la tecnología y la ciencia. La herramienta fue pensada para que programadores la instalen en sus aplicaciones y así ofrecer a los usuarios finales la posibilidad de generar visualizaciones sin tener conocimiento de cómo se construyen internamente. La misma permite crear visualizaciones partiendo de información provista por el sitio visitado. Los usuarios se vuelven responsables de la información que quieren extraer y visualizar, y de qué manera.

Palabras Claves

Visualización de información, Visualización, Programación, Usuarios finales, Aplicaciones web, Interfaz de usuario, Gráficos, Ciencia ciudadana, Interacción de usuarios, Biblioteca

Conclusiones

Una de las intenciones con el trabajo realizado en esta tesina es compartir conocimiento. No todos los desarrolladores pueden implementar cada funcionalidad que podría ser útil para los usuarios finales. La interacción con tecnologías que se desconocen lleva de mucho esfuerzo y tiempo, lo cual es muy valioso y la mayoría de las veces hace falta. A su vez, eso que algunos desarrolladores pueden desconocer, hay muchos otros que tienen conocimiento sobre cómo implementarlo. Al dejar a disposición de los usuarios finales un grupo de operaciones disponibles, queda bajo su responsabilidad la creación de visualizaciones. Esto hace que la interacción con los sitios sea dinámica. Los mismos no dependen de que sus administradores tengan que actualizarlo implementando nuevos tipos de visualizaciones, dicha implementación está provista por la herramienta implementada y visualizarlas queda en responsabilidad de los usuarios.

Trabajos Realizados

- Investigación y aprendizaje tanto de gráficos como de creación y exportación de bibliotecas.
- Implementación de distintos tipos gráficos, de un extractor de información y operaciones de transformación partiendo de tablas de información.
- Desarrollo de una interfaz gráfica que integra las mencionadas implementaciones.
- Implementación de un backend para volver las operaciones persistentes.
- Desarrollo y publicación de una herramienta extensible y re-utilizable combinando las implementaciones mencionadas.

Trabajos Futuros

- *Extender la herramienta:* crear nuevas formas de reconocer tablas de información y extraer información de las mismas; crear nuevas operaciones de transformación de tablas; agregar nuevos tipos de gráficos.
- *Mejoras generales en la interfaz.* Agregar la posibilidad de agrupar las visualizaciones y mostrarlas de distintas formas. Sistema de comentarios y etiquetas en las visualizaciones para futuras búsquedas.
- *Flexibilidad para cargar nueva información.*
- *Posibilidad de descargar las visualizaciones.*
- *Adaptar la herramienta como una extensión del navegador para no depender de que programadores la incluyan en sus sitios para poder utilizarla.*

Visualización de información en la web
dirigida por usuarios finales

Alumna: Josefina Estevez de Urquiza

Directores: Sergio Firmenich y Diego Torres

Resumen

En el siguiente trabajo se implementa una herramienta de visualización para ser utilizada en aplicaciones web previamente existentes. Uno de los objetivos es que sea utilizada en aplicaciones dentro del marco de ciencia ciudadana. De esta manera pretende democratizar la información y simplificar el trabajo a la hora de mejorar interfaces, reduciendo la barrera tecnológica entre la tecnología y la ciencia. La herramienta fue pensada para que programadorxs la instalen en sus aplicaciones y así ofrecer a lxs usuarixs finales la posibilidad de generar visualizaciones sin tener conocimiento de cómo se construyen internamente. La misma permite crear visualizaciones partiendo de información provista por el sitio visitado. Lxs usuarixs se vuelven responsables de la información que quieren extraer y visualizar, y de qué manera.

Agradecimientos

Quiero agradecer a mi familia por haberme acompañado y motivado siempre. Particularmente a mi mamá, por criarme en libertad y apoyarme en cada camino que elija sin importar lo distinto de nuestras ideas.

A mis amigas y a mi hermana, por la complicidad y el amor incondicional, porque no sería quien soy sin la influencia de sus valores. Compañeras en la lucha por una sociedad más justa e igualitaria.

A la UNLP, por su condición de pública y gratuita, por darme la oportunidad de formarme como profesional y como persona entender la importancia de que todos tengamos la posibilidad de acceder a ella.

A Diego y a Sergio, por entenderme y guiarme con paciencia durante todo el proceso de desarrollo de este trabajo.

Y por último a BMAT, por decidir confiar en mi trabajo e impulsarme a cerrar este ciclo educativo.

Índice general

1. Introducción	10
1.1. Motivación	10
1.2. Contribuciones	11
1.3. Organización	12
2. Visualización de la Información	14
2.1. Introducción	14
2.2. Visualización	15
2.3. Interacción	16
2.4. Taxonomía de visualizaciones	18
2.4.1. Unidimensional	18
2.4.2. Bidimensional	19
2.4.3. Tridimensional	20
2.4.4. Temporal	21
2.4.5. Multidimensional	22
2.4.6. Árbol	28
2.4.7. Red	29

3. Ciencia Ciudadana	31
3.1. Ciencia Ciudadana	31
3.2. Tecnología y ciencia	32
3.3. Participación ciudadana	32
4. Trabajo Relacionado	34
4.1. Cuscus	34
4.2. VizMe	36
4.3. ManyEyes	37
4.4. Google Chart Tools	38
4.5. Tableau	40
5. Desarrollo Propuesto	42
5.1. Arquitectura	43
5.2. Instalación y uso	58
5.3. API	62
5.4. Tecnologías	67
6. Evaluación	70
6.1. Encuesta inicial	72
6.1.1. Información demográfica	72
6.1.2. Relación con la tecnología	77
6.2. Tareas propuestas	84
6.3. Encuesta Final	85
6.3.1. Respecto a las tareas propuestas	86
6.3.2. Usabilidad	87
6.3.3. Feedback	89

6.4. Conclusiones del capítulo	92
7. Conclusiones y trabajo futuro	94
7.1. Conclusiones	94
7.2. Trabajo Futuro	95

Índice de figuras

2.1. Lista de escritorxs ordenada alfabéticamente	19
2.2. Cartograma	20
2.3. Tetera dibujada mediante gráfico 3D	21
2.4. Línea de tiempo	22
2.5. Gráfico de torta	23
2.6. Gráfico de barras	24
2.7. Gráfico de línea	25
2.8. Gráfico de área	26
2.9. Diagrama de dispersión	27
2.10. Gráfico de burbujas	28
2.11. Visualización tipo Árbol	29
2.12. Visualización tipo Red	30
4.1. Cuscus	35
4.2. VizMe	37
4.3. Google Chart Tools	38
4.4. Tableau	41
5.1. Paquetes	43

5.2. Clase Extractor	45
5.3. Clase NiceTable	46
5.4. Clase Visualization	47
5.5. Crear gráfico UI - Paso 1	48
5.6. Crear gráfico UI - Paso 2	49
5.7. Clase ChartVis	50
5.8. Filtrar columnas UI	54
5.9. Renombrar columnas UI	55
5.10. Cambiar tipo de datos de las columnas UI	56
5.11. Clase Transformation	57
6.1. Pagina web evaluación	71
6.2. Encuesta Inicial - Nacionalidad	73
6.3. Encuesta Inicial - Género	74
6.4. Nivel de educación	75
6.5. Encuesta Inicial - Rubro	76
6.6. Encuesta Inicial - Estado Profesional	77
6.7. Encuesta Inicial - Frecuencia del uso de internet	78
6.8. Encuesta Inicial - Frecuencia del uso de internet sin contar redes sociales	79
6.9. Encuesta Inicial - Frecuencia del uso del navegador	80
6.10. Encuesta Inicial - Frecuencia del uso de hojas de calculo	81
6.11. Encuesta Inicial - Experiencia creando gráficos a través de hojas de cálculo	82
6.12. Encuesta Inicial - Experiencia creando gráficos a través de hojas de cálculo	83

6.13. Encuesta Inicial - Experiencia con herramientas de manipulación de tablas y gráficos	84
6.14. Encuesta Final - ¿Pudieron completar todas las tareas propuestas?	86
6.15. Encuesta Final - Tiempo en realizar las tareas	87
6.16. Formulario SUS	89

Listings

4.1. Inicializar gráfico de torta en Google Chart Tools	38
5.1. Formulario para crear visualización de tipo mapa	51
5.2. Tipos de datos disponibles	52
5.3. Modelos del backend	58
5.4. Instalar TableVis	59
5.5. Importar TableVis en proyecto vue	59
5.6. Importar TableVis sin vue	59
5.7. Inicializar TableVis en proyecto vue	59
5.8. Inicializar TableVis sin vue	60

Índice de cuadros

5.1. API: Obtener Tabla	63
5.2. API: Crear tabla	64
5.3. API: Listar gráficos	65
5.4. API: Crear gráfico	66
5.5. API: Eliminar gráfico	67

Capítulo 1

Introducción

1.1. Motivación

Esta tesina trabaja sobre visualización de información en el marco de ciencia ciudadana. A continuación se definen brevemente estos dos conceptos, los cuales se amplían en los próximos capítulos.

Por un lado, la visualización de información es el conjunto de representaciones visuales de datos en orden de ayudar a las personas a interpretarlos y trabajar con ellos de manera más eficiente[1]. Por otro, se entiende por ciencia ciudadana a la participación de lxs ciudadanxs, que sin tener conocimientos científicos, colaboran en proyectos de carácter científico[2].

Para hacer efectiva la participación de lxs ciudadanxs en proyectos científicos, es esencial que se involucre a la tecnología. Para así simplificar a gran escala las tareas automatizando gran parte de ellas. Apoyar al crecimiento de la ciencia ciudadana es por ende apoyar al desarrollo de otras disciplinas. La visualización de información, a través de la representación gráfica, posibilita

el descubrimiento de patrones con ayuda de interacción y exploración visual. Las visualizaciones facilitan la comprensión y el análisis de grandes volúmenes de datos. Particularmente en el ámbito científico, visualizar fenómenos e información compleja se ha hecho muy útil al momento de analizarlos. Para lograr visualizaciones, es necesario tener conocimientos específicos, por lo que se necesitan personas profesionalmente formadas. Se requiere de una manipulación apropiada de la información para lograr representaciones. Gran cantidad de sitios web muestran información en formato tabla ya que es lo más simple de realizar. Otras representaciones gráficas requieren de conocimientos específicos y esto demanda mas trabajo y dependencia[3]. Lxs desarrolladorxs no pueden abarcar cada una de las necesidades que demandan lxs usuarixs. Cada unx prioriza algunas funcionalidades y sus implementaciones están limitadas por el tiempo y conocimiento que cada unx disponga. Hay muchos sitios web a comparación a la cantidad de programadorxs. Cada sitio debe ser modificado individualmente cuando quiera ser actualizado. Con la herramienta propuesta se pretende reducir esta barrera. Se brinda a lxs programadorxs la posibilidad de ofrecer a lxs usuarixs manipulación de tablas y creación de gráficos en sus sitios sin tener que programar estas funcionalidades. Esta tesina de este modo pretende reducir la problemática mencionada. Una herramienta de este tipo mejora muchos sitios al mismo tiempo[4].

1.2. Contribuciones

En esta tesina se implementa una herramienta de visualización de información, la cual podrá ser utilizada en distintos proyectos con el fin de

proporcionar visualizaciones de datos y así facilitar la comprensión de los mismos.

La misma deberá ser incorporada por usuarios expertos en sus aplicaciones. De esta manera le permiten a los usuarios finales la posibilidad de crear visualizaciones sin la necesidad de tener conocimientos previos en programación. La intención es que se incorpore en aplicaciones previamente realizadas en el marco de ciencia ciudadana cuya funcionalidad principal es facilitar la interacción entre requisitos científicos e ingresos de muestras. La instalación de la librería desarrollada permitirá a los usuarios que interactúen con las aplicaciones decidir sobre qué variantes y de qué manera visualizar la información, brindando posibilidad de seleccionar y filtrar datos para obtener como resultado visualizaciones acorde a sus necesidades. La herramienta les permite a los usuarios finales cambiar dinámicamente la forma en que se visualizan los datos dentro de las aplicaciones que la utilicen. Se elimina la necesidad de esperar a que la o las personas a cargo de la administración de las respectivas aplicaciones decidan qué visualizaciones mostrar.

1.3. Organización

En cuanto a la estructura del documento, en los primeros capítulos se explican algunos conceptos teóricos básicos sobre *Visualización de la información* y *Ciencia Ciudadana*. Luego, en el capítulo de *Trabajos Relacionados*, se hace mención a herramientas y trabajos que abarcan problemáticas similares, comparando en qué puntos se parecen y mencionando sus falencias. En el capítulo *Desarrollo Propuesto*, se habla de la herramienta desarrollada en

sí, explicando su arquitectura, cómo instalarla, cómo incluirla en proyectos, cómo usarla, y las tecnologías que utiliza. Luego, en el capítulo *Evaluación* se realiza un experimento a un grupo de personas analizando su interacción con la herramienta. Por último está el capítulo de *Conclusiones y trabajo futuro*.

A continuación, la herramienta desarrollada, será referida durante todo el documento utilizando el nombre de **TableVis**.

Este documento hace uso del lenguaje de forma no binaria, es decir que no hace referencia al género de las personas y por esto el uso de la 'X' estará en el recorrido de los capítulos. Se entiende al lenguaje como algo moldeable y flexible y se lo utiliza como espacio de disputa para resignificar nuestras prácticas. Se toma esta decisión intentando dar visibilidad a los cambios lingüísticos que traducen una nueva ideología y comprensión de la realidad[5]. Esta modalidad ya ha sido reconocida por otras facultades de la UNLP por resolución como por ejemplo la FaHCE¹.

¹Resolución Nro. 2086/17: Modificación del artículo 41 del Régimen de Enseñanza y Promoción para reconocer el uso plurales y dinámicos del lenguaje

Capítulo 2

Visualización de la Información

2.1. Introducción

La información, de distintos tipos y en distintos formatos, es parte de la cotidianidad de las personas. La falta de información no es un problema, el problema es como extraer conocimiento valioso de toda la información que tenemos disponible. En orden de procesar, entender, y poder extraer real conocimiento de la información, es importante la manera en la que se utiliza para representar la masiva cantidad de datos existente[6][7].

El objetivo básico de las visualizaciones es crear representaciones visuales que le permitan a las personas explotar sus capacidades a la hora de enfrentar la comprensión de enormes y complejas cantidades de información[7].

2.2. Visualización

La visualización es una representación visual sobre algo: persona, objeto, etc. Mediante las visualizaciones tenemos la posibilidad de transmitir ideas complicadas de manera más precisa, clara y eficiente, mostrando realmente lo que sea pertinente. La visualización tiene como objetivo ampliar la percepción del ser humano facilitando la adquisición del conocimiento. A la hora de tener que representar grandes cantidades de información, la visualización es utilizada para poder lograr esto de manera que tenga un sentido para el receptor. De la mano del gran crecimiento de las tecnologías y de internet, los volúmenes de datos cada vez son mas y han surgido nuevas tendencias de visualizaciones[7][8].

Visualización de la información

La visualización de la información surge y se desarrolla a partir de los 90 con motivaciones de extender la visualización de lo real a lo abstracto, es decir, a través de fenómenos físicos, plasmar información no física. Mediante la visualización de la información, al reorganizarla, se pueden descubrir patrones que no están a simple vista y relacionar la información. Entre el 2004 y el 2005 emana de la visualización de la información la nueva iniciativa del análisis de datos (visual analytics), con el eslogan “Detectar lo esperado, Descubrir lo inesperado” [9]. La visualización de la información amplía la capacidad cognitiva y de percepción. Es la representación gráfica de la información donde se facilita la comparación, la detección de cambios y reconocimiento de patrones, entre otras habilidades cognitivas. Ayuda a las personas a entender los datos y analizarlos. Los métodos de visualización

hacen que la masiva cantidad de datos que tenemos pueda ser aprovechada. No nos sirve de mucho estar sobrados de información si nos es muy complejo procesarla. Se necesitan de diversos campos para colaborar con los grandes desafíos que presenta la visualización de la información, entre ellos está la ciencia de la computación, la cual facilita el entendimiento y la interpretación a través de diversas herramientas[7][10].

Visualización y Conocimiento

El conocimiento es la información que hemos adquirido y procesado a través de procesos mentales. Para llegar a crear conocimiento es necesaria la interacción con la información para una asimilación de la misma. Por lo tanto, podemos considerar que la visualización de la información es una técnica que aplicamos para adquirir conocimiento[9][8].

2.3. Interacción

La interacción es elemental a la hora de crear herramientas de visualización. Cuando la cantidad de información es muy grande, no es posible mostrarla toda de una sola vez. Los usuarios, a través de su interacción, generan cambios en las visualizaciones decidiendo que aspectos de la información quiere visualizar[8]. Para la visualización de la información, las técnicas de interacción son indispensables. El objetivo de la visualización de la información es que el usuario pueda efectivamente comprender la información incorporando el conocimiento, y para lograrlo, resulta indispensable que el usuario manipule las visualizaciones[1][11]. A continuación se mencionan

algunas formas de interacción.

Filtrado y Selección

Se debe permitir la selección parcial de los datos, quedando así el usuario solamente con los que le sean de interés. Es de gran utilidad descartar la información que no nos interesa y quedarnos sólo con la pertinente para luego manipularla de distintas maneras en futuras oportunidades. A través de la selección podemos lograr un ajuste de nuestra información, disminuyendo su cantidad dejando solamente lo que sea necesario visualizar, y así poner el foco en la información que sea de interés[12].

Búsqueda

Cuando se tiene gran cantidad de información, puede ser una tarea difícil la de selección y filtrado. Utilizando el mecanismo de búsqueda se puede automatizar el modo de quedarse con la información deseada filtrando la que tenga coincidencias con algún patrón indicado[12].

Orden

La técnica del orden también nos permitirá encontrar la información de manera mas rápida indicando el campo que queramos ordenar y con que criterio[12].

2.4. Taxonomía de visualizaciones

En orden a representar y analizar información existen diversos métodos para visualizarla. Según las diferentes problemáticas que se intentan resolver por lxs usuarixs surgen diferentes taxonomías de visualizaciones. A continuación se describen distintos tipos provenientes de la taxonomía de visualizaciones propuesta en *Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations*[12].

2.4.1. Unidimensional

Las visualizaciones de una dimensión son lineares. Se organizan de manera secuencial y están contenidas sólo por texto[13]. Podría tratarse de código de un programa, o listas ordenadas alfabéticamente. Es difícil para unx usuari detectar patrones en este tipo de visualizaciones, por ejemplo encontrar los elementos que coincidan con determinado atributo, o filtrarlos por alguna característica[12]. La siguiente imagen corresponde a un ejemplo de visualización de una dimensión, es una lista de escritorxs ordenada alfabéticamente.

G. Egido, Luciano	G. Martin, Benjamin
G. Mazzucco, Melania	G. Payne, Stanley
G. Reigosa, Carlos	G. Tierney, William
Gaarder, Jostein	Gabilondo, Ángel
Gabriel, Markus	Gabriel (ed.), Pere
Gabriel y Galán, José Antonio	Gacinski, Sylvianne A
Gadamer, H.-G.	Gadamer, Hans-Georg
Gaddis, John Lewis	Gaitán, Fernando
Gala, Antonio	Galán, Francisco
Galanaki, Rhea	Galarza, Sergio
Galbraith, James K.	Galbraith, Robert
Galeano, Eduardo	Galison, Peter
Gallego, Ferran	Gallego, Ferrán
Gallego, Ferran	Gallego, José Luis
Gallego, Rubén	Gallego (ed.), Ferran
Gallego Roca, Miguel	Gallegos, Raúl
Gallud Jardiel, Enrique	Galmés de Fuentes, Álvaro
Galmés de Fuentes (ed.), Álvaro	Gálvez, Pedro
Gambetta, Diego	Gamoneda, Amelia
Gamoneda, Antonio	Gándara, Alejandro
Garci, José Luis	García, Hugo

Figura 2.1: Lista de escritorxs ordenada alfabéticamente

2.4.2. Bidimensional

Las visualizaciones de dos dimensiones son planas o geo-espaciales. Cada elemento dentro de la misma esta representando una parte del total del área de la visualización[13]. Puede haber muchas capas para representar un mapa por ejemplo, pero cada una de las mismas sera de 2 dimensiones. Ejemplos de este tipo de visualización pueden ser mapas geográficos o planos de

plantas[12]. La siguiente imagen corresponde a un cartografía, la cual es un ejemplo de visualización de dos dimensiones.

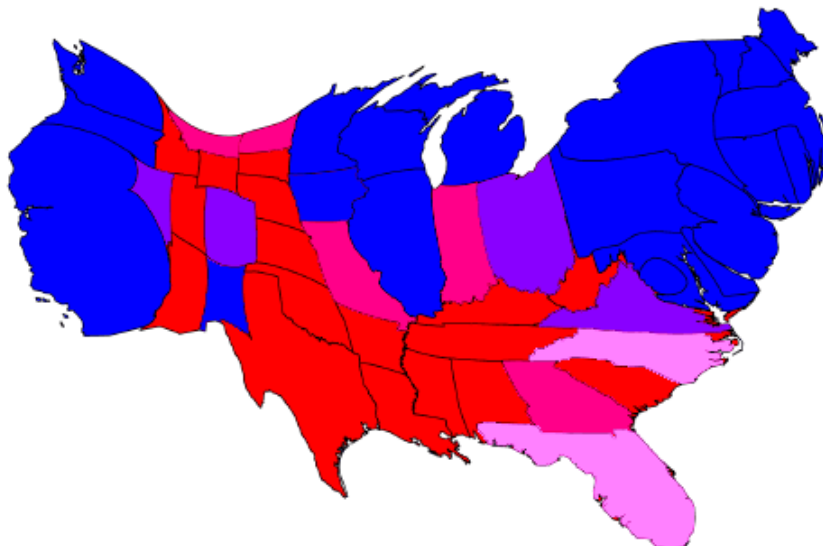


Figura 2.2: Cartograma

2.4.3. Tridimensional

Las visualizaciones de tres dimensiones nos ayudan a proyectar objetos del mundo real permitiéndole ver a los usuarios las relaciones tridimensionales de los mismos y así poder comprender sus posiciones y orientaciones[12]. Son de uso frecuente en las ramas de la arquitectura y la medicina. La siguiente imagen, corresponde a una visualización 3D de una tetera, ejemplo de visualización tridimensional.

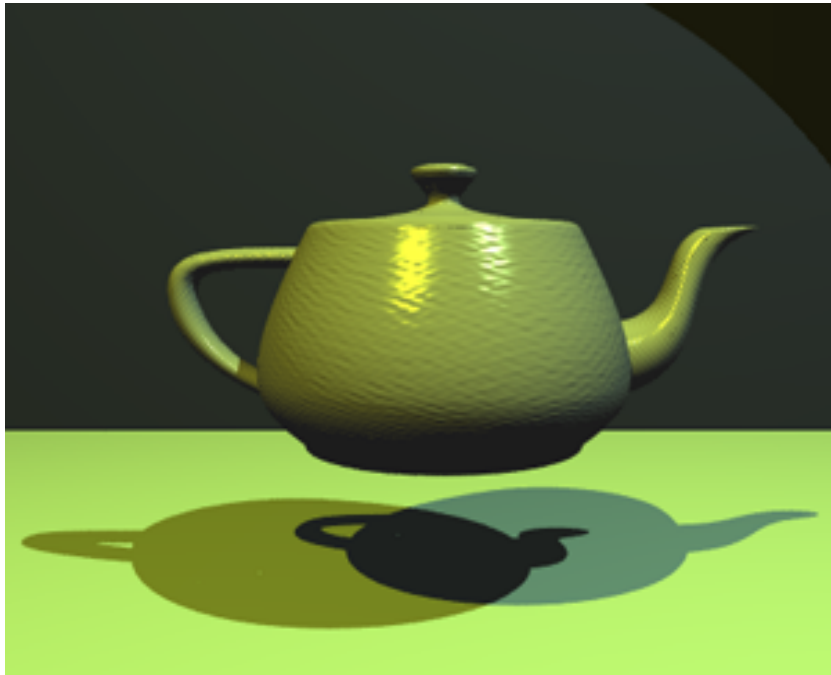


Figura 2.3: Tetera dibujada mediante gráfico 3D

2.4.4. Temporal

La distinción en este tipo de visualizaciones es que tienen un atributo relacionado a un periodo de tiempo, el cual los posiciona de manera ordenada. Esto permite buscar elementos antes, después, o durante determinado momento[13]. Un ejemplo de uso puede ser en historiales médicos. La siguiente imagen corresponde a una visualización de línea del tiempo, que es una visualización de tipo temporal.

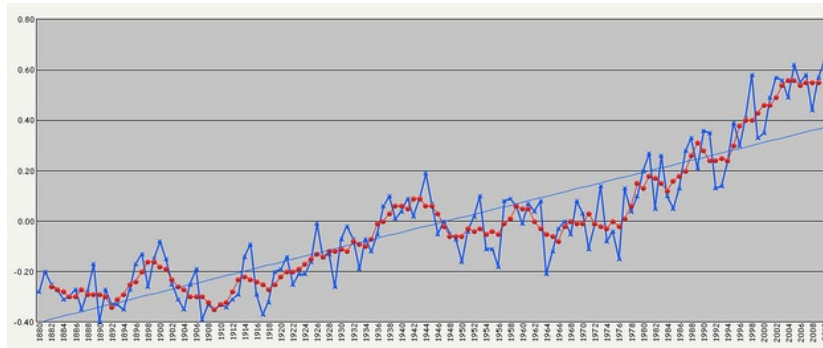


Figura 2.4: Línea de tiempo

2.4.5. Multidimensional

Los elementos en las visualizaciones multidimensionales están compuestos por n atributos lo cual se conlleva a n dimensiones[12]. A continuación se describen ciertos ejemplos, algunos de los cuales están incluidos en el desarrollo propuesto.

Gráfico de torta

Este tipo de gráfico se divide en sectores, y cada sector es un porcentaje del total del gráfico. Por lo tanto, sin importar la cantidad de sectores que tengamos, la suma de las proporciones de cada uno va a dar un total de 100. Es útil para comparar dichos sectores entre sí. La siguiente imagen corresponde a un gráfico de torta, mostrando los porcentajes de los diferentes valores posibles para el campo[14].

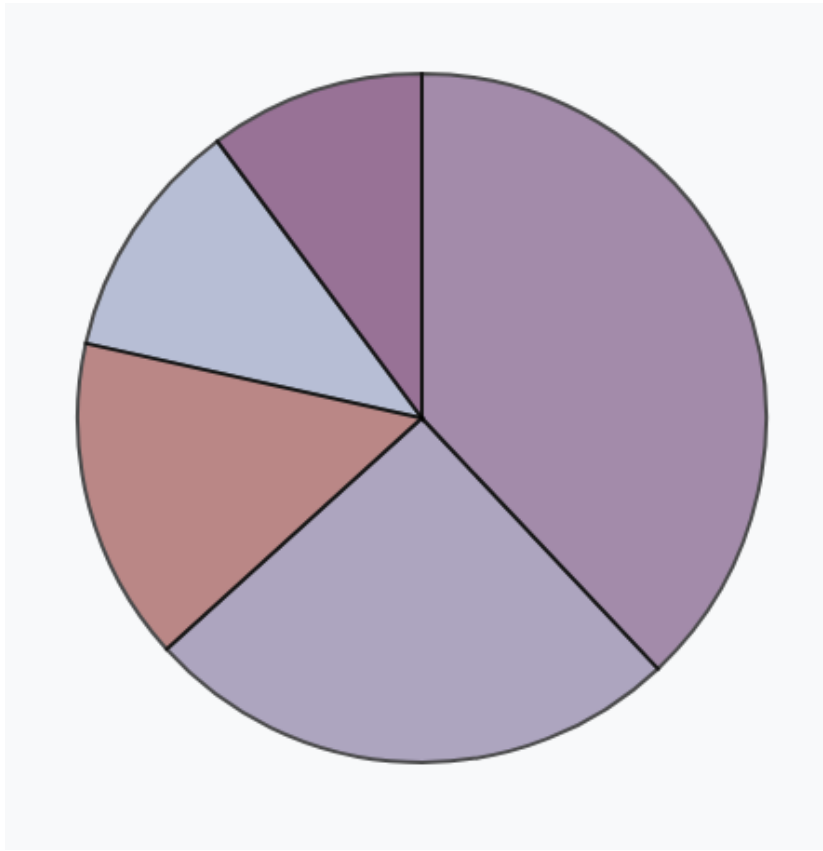


Figura 2.5: Gráfico de torta

Gráfico de barras

Representa la información en forma de barras, cuya longitud está relacionada con su valor. Las barras pueden encontrarse de manera vertical o de manera horizontal, funcionando exactamente del mismo modo salvando la diferencia de la orientación de las mismas. Se utiliza para representar valores no continuos[14]. La siguiente imagen corresponde a un gráfico de barras verticales. Muestra una barra por cada valor posible para el campo, mostrando

sus etiquetas en el eje x y su valor numérico expresado por el alto de las barras sobre el eje y.

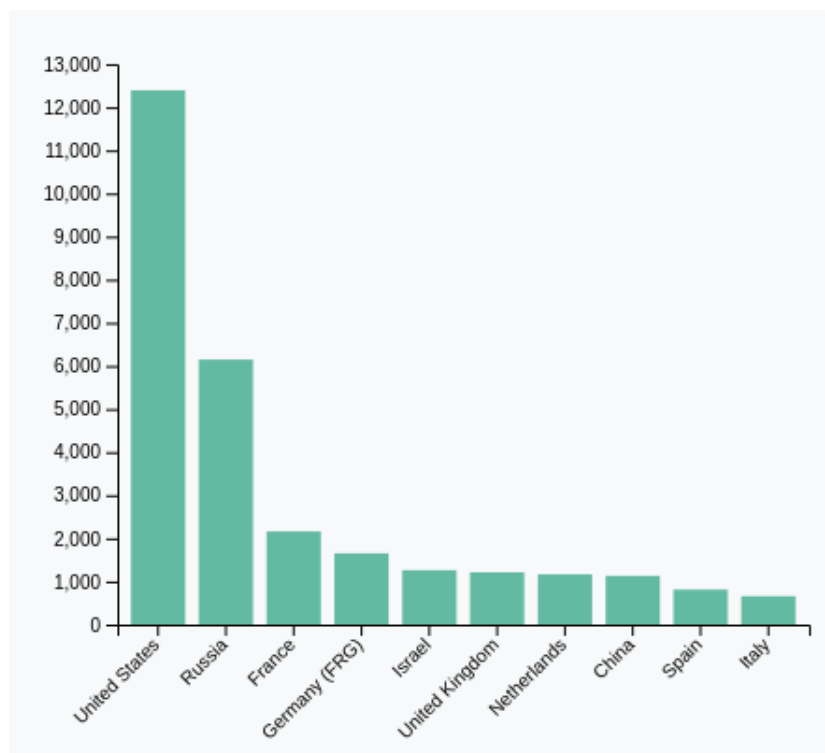


Figura 2.6: Gráfico de barras

Gráfico de línea

Este tipo de visualización nos permite representar ciertos valores relacionándolos con determinados puntos. Se usa mucho para representar tendencias durante el tiempo. Existe una variación del mismo donde se pueden representar distintos valores relacionados con determinados puntos, conocido como gráfico multilínea[14]. La siguiente imagen corresponde a un gráfico de

línea.

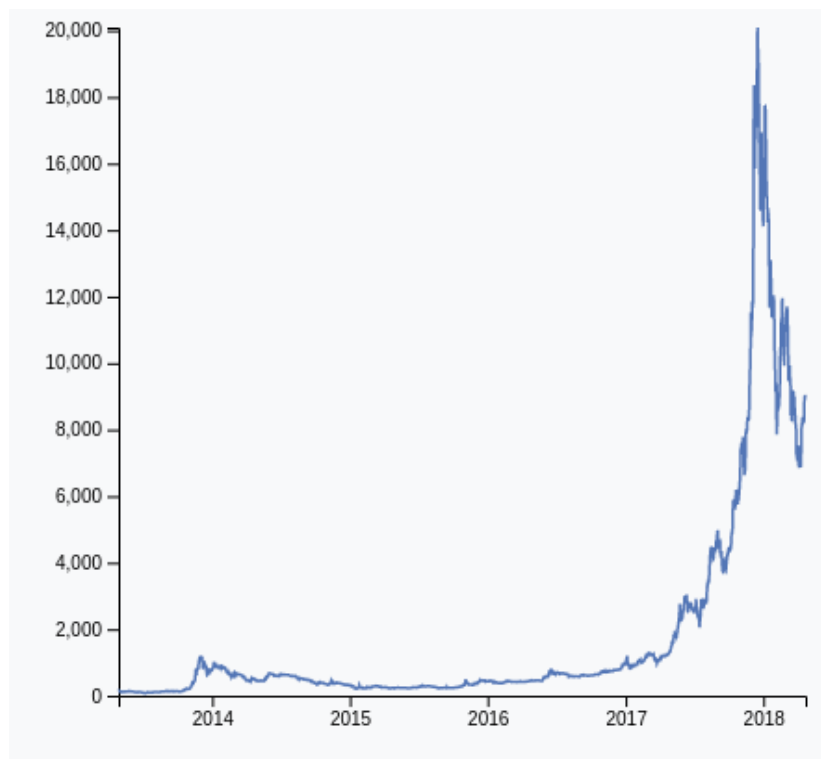


Figura 2.7: Gráfico de línea

Gráfico de área

Representa la información clasificándola en áreas. Las mismas se determinan a partir de un gráfico lineal y coloreando el área por debajo de la línea[14]. La siguiente imagen es un ejemplo de este tipo.

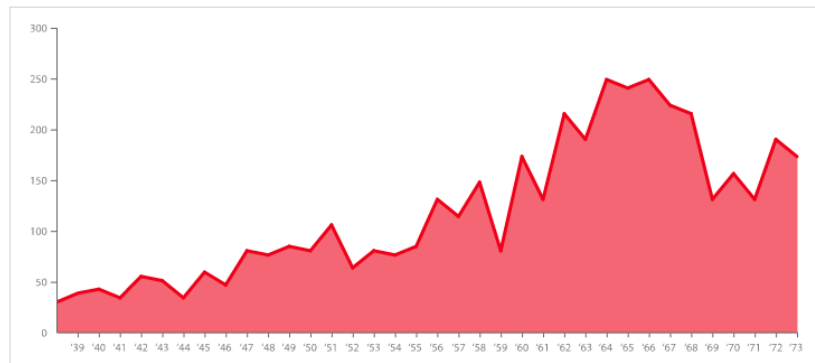


Figura 2.8: Gráfico de área

Diagrama de dispersión

Visualiza la información en coordenadas cartesianas. Cada punto muestra la relación entre dos variables siendo una la distancia horizontal y la otra la vertical desde los ejes[14]. A continuación podremos observar una imagen con un diagrama de dispersión de ejemplo, el cual contiene un eje x, un eje y, y puntos representando relaciones entre los mismos.

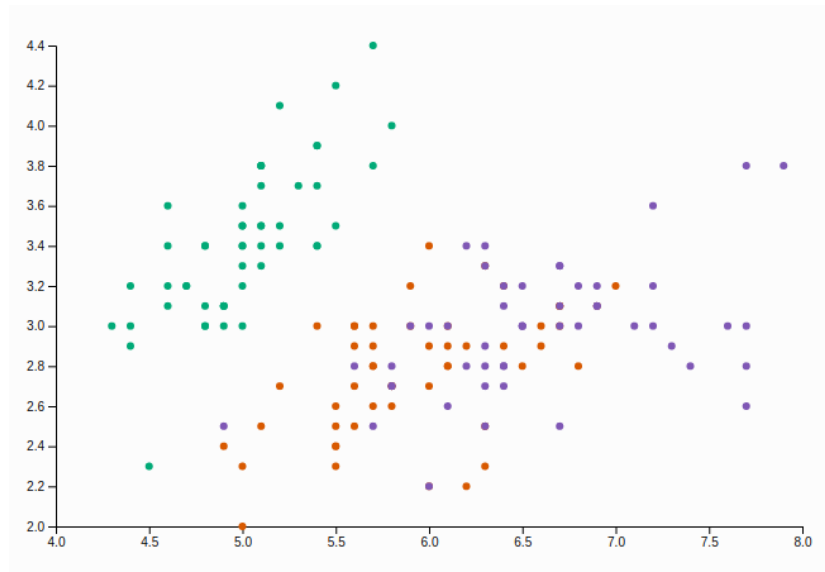


Figura 2.9: Diagrama de dispersión

Gráfico de burbujas

Es una variación del diagrama de dispersión. Cada elemento dentro de este gráfico está compuesto por dos valores. Por un lado el posicionamiento que el elemento debe tener en el gráfico, el punto horizontal y el punto vertical donde debe ubicarse. Por otro lado el valor que determinará el tamaño de la burbuja en el gráfico[14]. La siguiente imagen es un ejemplo de un gráfico de burbujas.

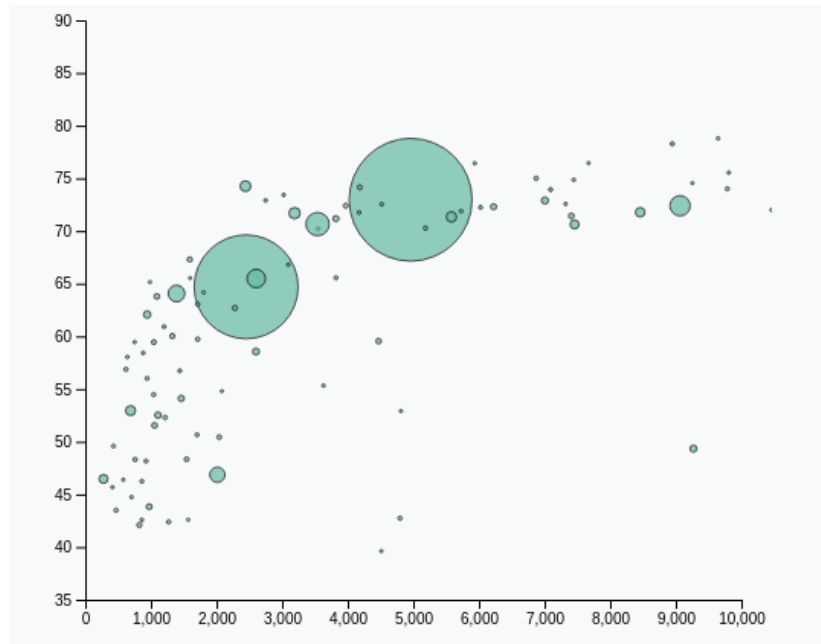


Figura 2.10: Gráfico de burbujas

2.4.6. Árbol

Los elementos en este tipo de visualizaciones tienen jerarquías, es decir que un elemento va a tener relacionado un elemento hijo o/y un elemento padre[12]. La siguiente imagen corresponde un ejemplo de una visualización de tipo árbol.

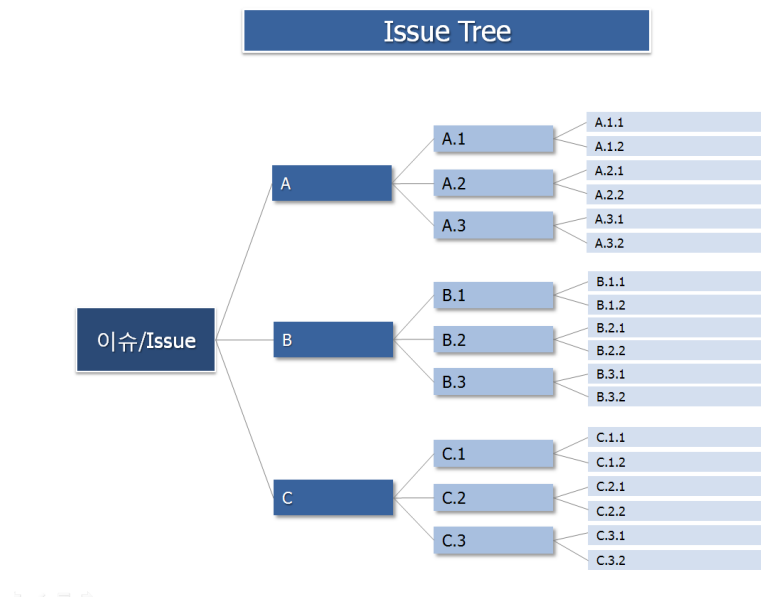


Figura 2.11: Visualización tipo Árbol

2.4.7. Red

A veces, para marcar las relaciones entre elementos no alcanza con una estructura jerárquica como la mencionada de tipo árbol y es necesario marcar relaciones de un elemento con mas de otro elemento[12]. La siguiente imagen corresponde un ejemplo de una visualización de tipo red.



Figura 2.12: Visualización tipo Red

Capítulo 3

Ciencia Ciudadana

3.1. Ciencia Ciudadana

Se entiende al concepto de ciencia ciudadana como la participación de ciudadanxs a la hora de generar datos para alcanzar resultados científicos. Lxs ciudadanxs, sin la necesidad de tener conocimientos científicos específicos, pueden aportar sus recursos y participar en proyectos de carácter científico. La colaboración en estos proyectos es de utilidad social, con la posibilidad de resultar en deducción de teorías y descubrimientos[15][16].

Este concepto de la participación de la ciudadanía para la recolección de información no es algo nuevo, uno de los registros más antiguos es de comienzos del siglo XIX. Es el caso de los estudios de las mareas oceánicas, que resultaban impredecibles, haciendo que la navegación fuera peligrosa. Si bien había gente investigando sobre la problemática, resultaba muy complejo establecer algún tipo de patrón. Hasta que se organizó un proyecto para estudiar las mareas, el cual implicó la participación de miles de personas

a lo largo de cientos de distintas locaciones, y gracias al compromiso de la ciudadanía, pudieron sacarse conclusiones para predecir el comportamiento de las mareas y así evitar accidentes durante la navegación[2].

3.2. Tecnología y ciencia

De la mano de los avances de la tecnología, por un lado la barrera comunicacional se redujo, pudiendo tener un constante acceso a la información. Por otro lado, con la evolución de los recursos de computación, la capacidad de almacenar información también se incrementó permitiendo almacenar grandes volúmenes de datos[10].

Se conoce como ciberciencia al uso de las TIC¹ con objetivos científicos dentro de proyectos de diversas disciplinas. Además del uso de las TIC, la ciberciencia se caracteriza por el acceso del público a este tipo de proyectos, y por lo tanto a su participación dentro de los mismos[2].

La aparición de la tecnología como contribución a la ciencia, no sólo es una forma para que lxs ciudadanxs participen dentro de la misma, sino que también otorga una posibilidad de registro de una gran cantidad de datos que sería posible que se escapen si quedaran sólo a cargo de lxs científicxs.

3.3. Participación ciudadana

La participación de lxs ciudadanxs en la ciencia ciudadana a través del uso de la tecnología puede tener distintos niveles de compromiso dentro de

¹Tecnologías de la información y la comunicación

los proyectos. Puede ser que su participación sea meramente la recolección de datos, sin más vinculación dentro del proyecto, o puede ser que están más estimuladxs a la participación y se lxs coloque en un lugar más decisivo dentro del proceso. En la mayoría de los casos, lxs ciudadanxs involucradxs son voluntarixs y sus motivaciones pueden ser diversas, pueden sentirse interesadxs en la temática que se investiga o puede que su intención sea colaborar a mejorar su entorno ya que muchos proyectos tienen carácter ambiental. La participación en investigaciones puede significar a una colaboración concreta en la cotidianidad de las personas y sus comunidades[16].

Capítulo 4

Trabajo Relacionado

4.1. Cuscus

Cuscus es una herramienta de visualización de la información. Cuscus es una herramienta que le permite a los usuarios finales crear visualizaciones partiendo de una matriz de información. Provee a estos usuarios la posibilidad de crear visualizaciones según lo que les sea conveniente y además sin necesidad de tener conocimientos técnicos sobre cómo crearlas. Es decir que, sin una herramienta facilitadora como esta, no sería posible generarlas. Cuscus es una herramienta en un contexto de una hoja de cálculo, que agrega la funcionalidad de generar visualizaciones utilizando un editor de dibujo. Por un lado tiene la hoja de cálculo, que es muy similar a un Microsoft Excel¹, y por el otro, del estilo Adobe Illustrator² o Microsoft Powerpoint³, el editor gráfico[17]. La siguiente figura corresponde a una captura de pantalla de

¹Microsoft Excel <https://products.office.com/excel>

²Adobe Illustrator <https://www.adobe.com/illustrator>

³Microsoft Powerpoint <https://products.office.com/es-ar/powerpoint>

Cuscus, donde se ve tanto la sección de hoja de calculo como la del editor gráfico.

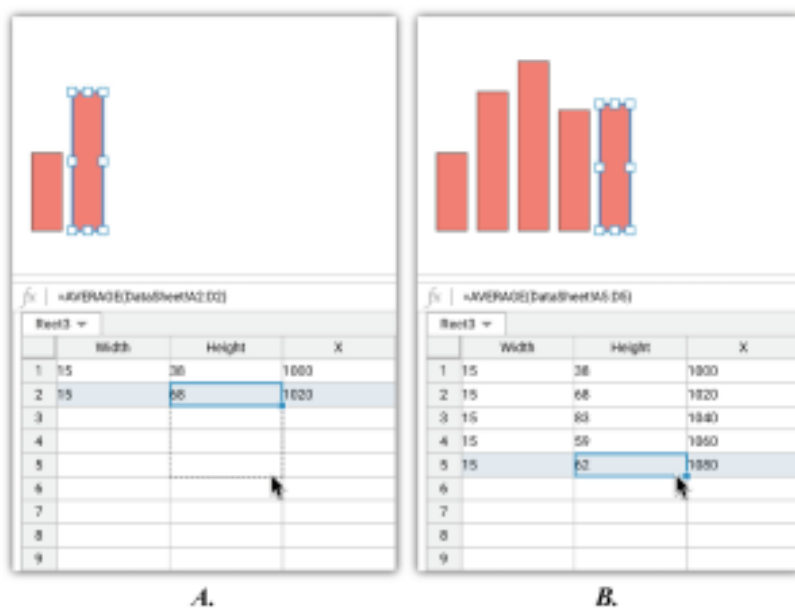


Figura 4.1: Cuscus

La primera desventaja es que, si bien una gran cantidad de personas sabrá utilizar el editor de dibujo (lo cual achica la brecha de requerimiento de conocimientos técnicos específicos para generar gráficos), de todas maneras requiere de conocimientos para hacerlo. Otra desventaja, es que cada usuario final deberá disponer de la herramienta para poder generar visualizaciones.

4.2. VizMe

VizMe es una extensión del navegador Firefox⁴. Con la instalación de la extensión se pondrá a disposición un panel que podrá mostrarse u ocultarse cuando sea necesario. Según la tarea que quiera realizarse se mostrarán distintas funcionalidades a disposición en el panel.

Una de las funcionalidades que brinda el panel es utilizar el buscador de google sin tener que salir del sitio. Esto es una ventaja ya que se elimina la necesidad de abrir una nueva ventana o pestaña para hacerlo y tener así que recordar lo que estábamos buscando, y nos evita también el tener que volver a la página anterior para relacionar la información.

Otra funcionalidad posible es la de recolectar información. Por ejemplo una persona llamada Jane está explorando un sitio y decide buscar noticias relacionadas con cierta información contenida en el mismo, entonces, a través del panel que ofrece VizMe ha encontrado algunas que le resultan interesantes, por lo que puede guardarla utilizando la interfaz de VizMe y de esta forma acceder a ella luego y mantener la relación[18].

VizMe le permite a lxs usuarixs agregar contenido al que ya es provisto por el sitio web y guardarlo. Permite agregar funcionalidad a la del sitio web interactuando con el mismo y así poder sacar provecho al analizar y utilizar la información. VizMe permite agregar contenido externo a cada sitio web y conservarlo para futuros usos. Otra ventaja es que lxs desarrolladorxs de los sitios no tuvieron que desarrollar estas funcionalidades.

La siguiente figura corresponde a una captura de pantalla de VizMe, donde se está realizando una búsqueda en google.

⁴<https://www.mozilla.org/>



Figura 4.2: VizMe

4.3. ManyEyes

ManyEyes es una aplicación web en la cual las principales actividades son las de subir datos, crear visualizaciones, y hacer comentarios tanto en los datos como en las visualizaciones[19]. ManyEyes funciona como una red social, cada elemento publicado tiene una sección de foro discusión donde los usuarios pueden hacer comentarios para compartir opiniones e información y debatir al respecto. El principal componente de ManyEyes son las tablas de información compuestas por filas y columnas. Cada columna tiene un tipo asociado. En ManyEyes, los usuarios son quienes cargan la información. Una desventaja que posee ManyEyes es que los usuarios deben estar registrados en el sitio para acceder al mismo, siendo esto una barrera para su uso.

4.4. Google Chart Tools

Google Chart Tools provee una serie de gráficos para ser utilizados por usuarios expertos en conocimiento web. Se utiliza por programadores a través de una API a la cual se le envía cierta configuración para obtener determinado gráfico y así incluir distintos tipos de visualizaciones en sus sitios web[20].

La siguiente imagen es un ejemplo de uno de los gráficos que ofrece la herramienta Google Chart Tools.

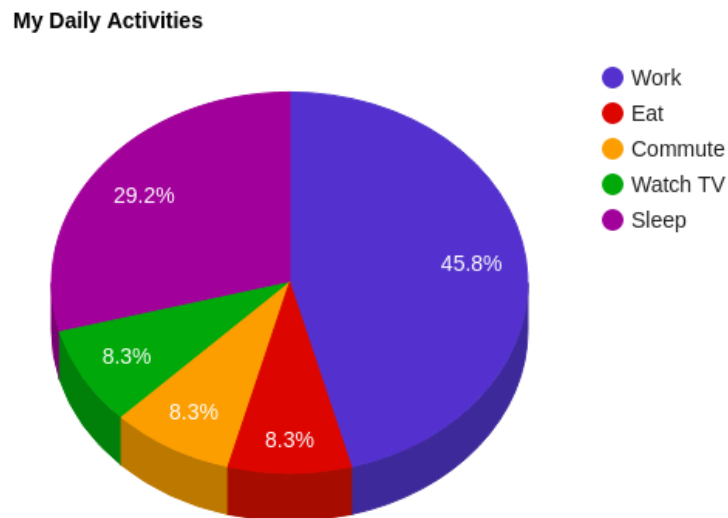


Figura 4.3: Google Chart Tools

El siguiente fragmento de código, muestra cómo instanciar el gráfico de la imagen anterior:

```
1 <html>  
2   <head>
```

```

3      <script type="text/javascript" src="https://www.gstatic.
      com/charts/loader.js"></script>
4      <script type="text/javascript">
5          google.charts.load("current", {packages:["corechart"
              ]});
6          google.charts.setOnLoadCallback(drawChart);
7          function drawChart() {
8              var data = google.visualization.arrayToDataTable([
9                  ['Task', 'Hours per Day'],
10                 ['Work',     11],
11                 ['Eat',      2],
12                 ['Commute',  2],
13                 ['Watch TV', 2],
14                 ['Sleep',    7]
15             ]);
16
17             var options = {
18                 title: 'My Daily Activities',
19                 is3D: true,
20             };
21
22             var chart = new google.visualization.PieChart(
23                 document.getElementById('piechart_3d'));
24             chart.draw(data, options);
25         }
26     </script>
27 </head>
28 <body>
29     <div id="piechart_3d" style="width: 900px; height: 500px
30         ;"></div>

```

```
29 </body>
30 </html>
```

Listing 4.1: Inicializar gráfico de torta en Google Chart Tools

Si bien esto provee una manera rápida de insertar visualizaciones, ya que tiene una curva de aprendizaje chica a comparación de librerías de visualizaciones como d3, está limitado a que lxs administradorxs del sitio incluyan específicamente cada visualización que necesiten.

4.5. Tableau

Tableau es una herramienta de visualización que permite a usuarixs finales crear visualizaciones. Dichas visualizaciones se crean seleccionando información y campos dentro de la misma. Le permite a lxs usuarixs ordenar información y filtrarla[20]. Tableau permite organizar las visualizaciones dentro de tableros y así mantenerlas ordenadas para su fácil acceso. Tableau permite a lxs usuarixs interactuar y crear visualizaciones sin tener conocimiento de cómo se implementan las mismas. Una desventaja acerca de Tableau es que se utiliza siempre en el marco de la misma aplicación.

La siguiente imagen corresponde a una captura de pantalla de Tableau, donde se está configurando una visualización, indicando las columnas y filas de interés.

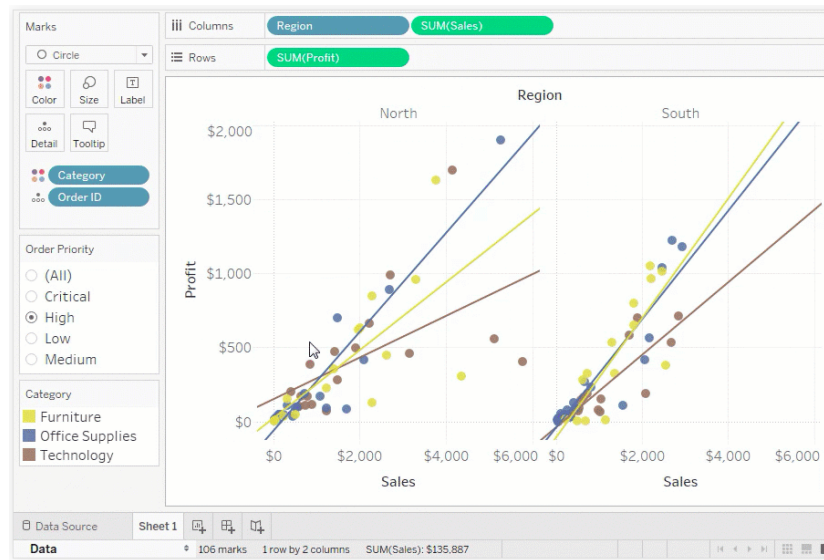


Figura 4.4: Tableau

Capítulo 5

Desarrollo Propuesto

El desarrollo propuesto se trata de una biblioteca javascript llamada **TableVis**. Dicha biblioteca es para que usuarios expertos la incluyan en sus aplicaciones web y así brindar nuevas funcionalidades a los usuarios finales. La misma, a partir de estructuras de información de tipo tabla, formadas por filas y columnas, permite generar diferentes visualizaciones.

La biblioteca, a partir del contenido de una tabla existente, genera una tabla con nuevas funcionalidades. Esta nueva tabla ofrece operaciones que no podían realizarse sin la biblioteca: filtrar columnas, renombrar columnas, elegir tipos para las columnas (esto permite buscar y ordenar dentro de la tabla según el tipo de columna que sea), paginación y la posibilidad de crear gráficos. La biblioteca puede ser incluida en cualquier aplicación web con el objetivo de brindar a los usuarios finales flexibilidad para la creación de diferentes visualizaciones. La misma se implementó junto con un backend, por lo que, las modificaciones en las tablas y los gráficos que se generen serán persistentes.

5.1. Arquitectura

TableVis esta compuesta principalmente por extractores, transformaciones y visualizaciones. Estos componentes operan y se relacionan entre sí con objetos del tipo **NiceTable**. La figura 5.1 corresponde al UML de paquetes de la herramienta. Luego se explican en detalle las clases dentro de los mismos y sus relaciones.

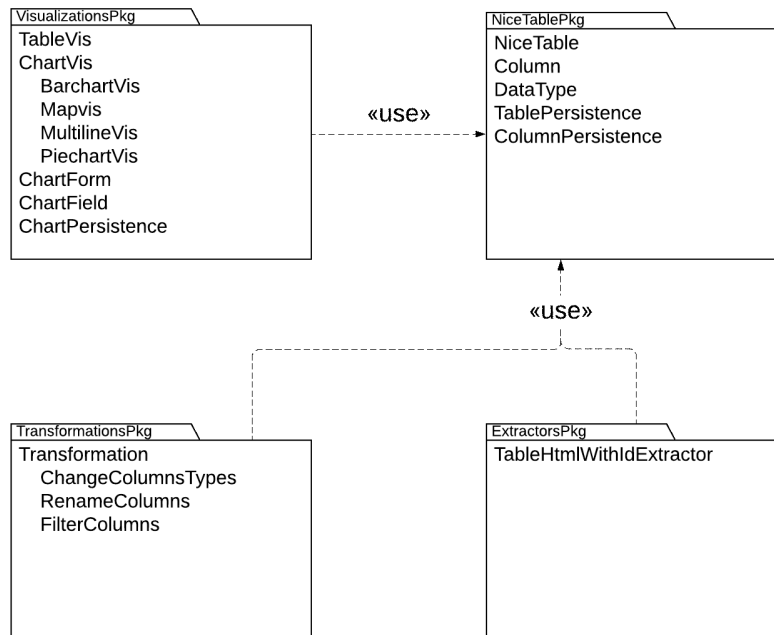


Figura 5.1: Paquetes

Extractores

Los extractores, se encargan de obtener información contenida en el DOM¹. Son capaces de reconocer la información contenida en una tabla para la creación de una nueva. Con la información obtenida, crean un objeto de clase NiceTable. Este objeto, incluye en él la información correspondiente a las columnas y filas de la tabla, de manera que su manipulación dentro de las visualizaciones sea conveniente. En el desarrollo propuesto, se implementó un extractor que obtiene información de una tabla a partir de un identificador. Extrae las filas y columnas de un elemento html con etiqueta table² a partir de un atributo id³.

La siguiente imagen corresponde al UML de clase Extractor. La misma podría extenderse implementando otro tipo de extractores.

¹Modelo de Objetos del Documento

²Estructura HTML para crear filas y columnas en una pagina web

³Identificador único de cada elemento dentro de una página HTML

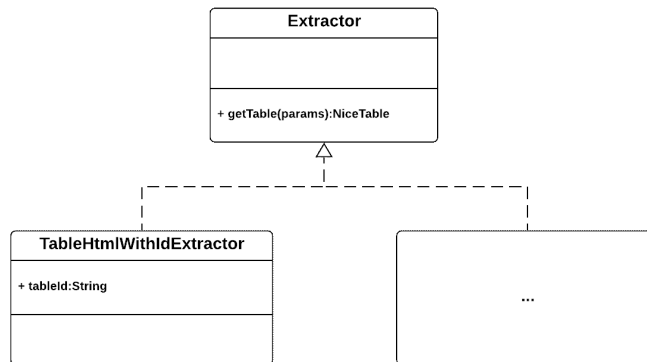


Figura 5.2: Clase Extractor

NiceTable

La figura 5.3 corresponde a la clase `NiceTable` presentada en el párrafo anterior. Los extractores devuelven objetos de este tipo, las visualizaciones los necesitan para visualizarse y las transformaciones se encargan de hacerles modificaciones, por lo que es el componente principal de la aplicación. Los objetos de esta clase tienen filas y columnas. Cada columna tiene: un título, que es el nombre con el que va a ser visualizada; un índice, que es la relación con la columna correspondiente de la tabla original; un tipo: indicando a que tipo de dato corresponde; y una propiedad booleana que indica si la columna debe ser o no mostrada. Cada una de las filas tiene el valor de cada una de las columnas correspondiente a la misma.

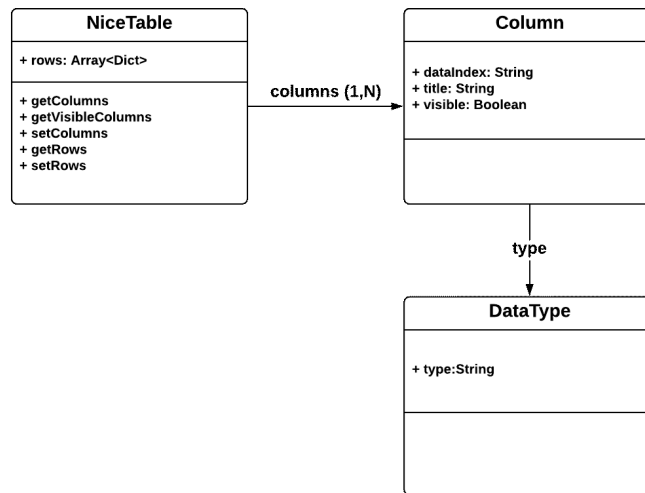


Figura 5.3: Clase NiceTable

Visualizaciones

Existen dos tipos de visualizaciones: de tipo tabla y de tipo gráfico. La figura 5.4 representa esta jerarquía.

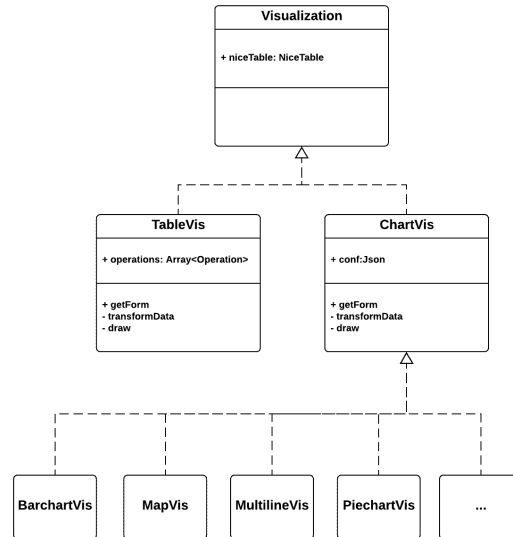


Figura 5.4: Clase Visualization

La visualización de tipo tabla, con el objeto NiceTable generado por el extractor, es capaz de visualizar una tabla con comportamiento más inteligente que el de una tabla original. Las visualizaciones de tipo gráfico, además del objeto NiceTable, necesitan de una configuración. Esta configuración indica si se incluyen total o parcialmente las filas, y especifica las columnas que deben tenerse en cuenta para la creación del mismo. Las figuras 5.5 y 5.6 corresponden al formulario que provee la interfaz para realizar esta acción. En la 5.5, se selecciona el tipo de gráfico que se quiere agregar.

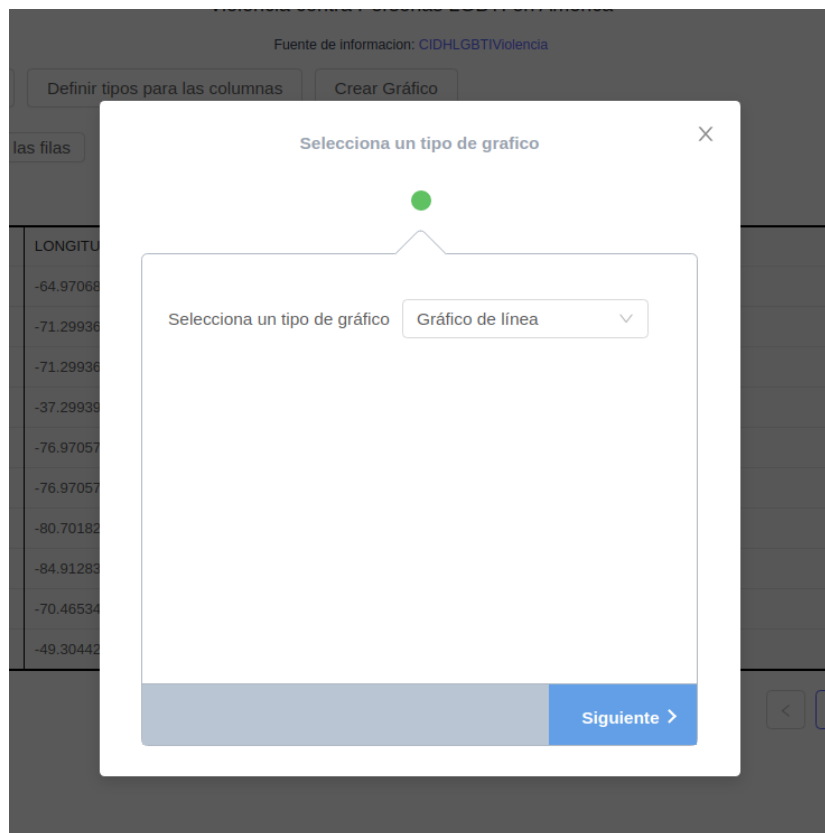


Figura 5.5: Crear gráfico UI - Paso 1

En la 5.6, habiendo seleccionado un tipo de gráfico linear, se solicita agregar las columnas correspondientes para la creación del mismo.

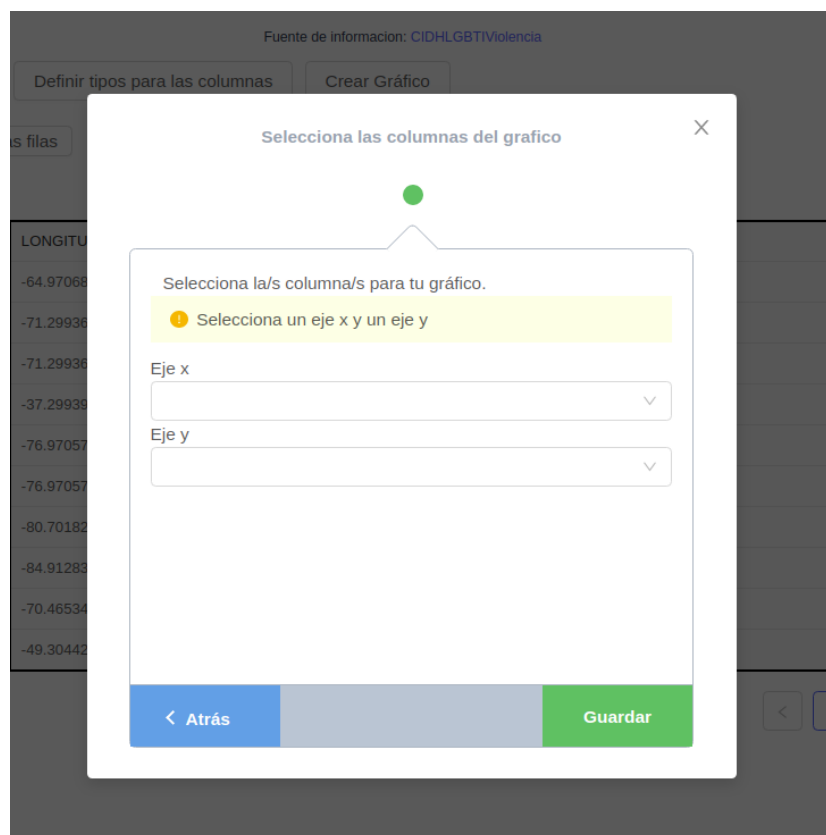


Figura 5.6: Crear gráfico UI - Paso 2

Las visualizaciones de tipo gráfico, parten de la información correspondiente a una tabla. Con el objeto NiceTable generado inicialmente por el extractor y la configuración, son capaces de dibujarse en el DOM. Cada una de estas visualizaciones sabe como dibujarse y lo hace a través de la interacción con una librería externa. Dentro de las visualizaciones, está implementada la comunicación que realiza con la librería que utilice para dibujarse. Por ejemplo, las visualizaciones Barchart, Piechart, ScatterplotChart y Linechart

usan d3 para hacerlo, mientras que Mapvis hace uso de la librería Leaflet⁴, la cual internamente utiliza OpenStreetMap⁵ para renderizar los mapas.

La figura 5.7 corresponde a la clase ChartVis, clase que representa a los gráficos. La misma podría seguir extendiéndose agregando nuevos tipos de gráficos. Para esto, debe tenerse en cuenta qué métodos deben implementarse para que su incorporación funcione de manera exitosa en la aplicación.

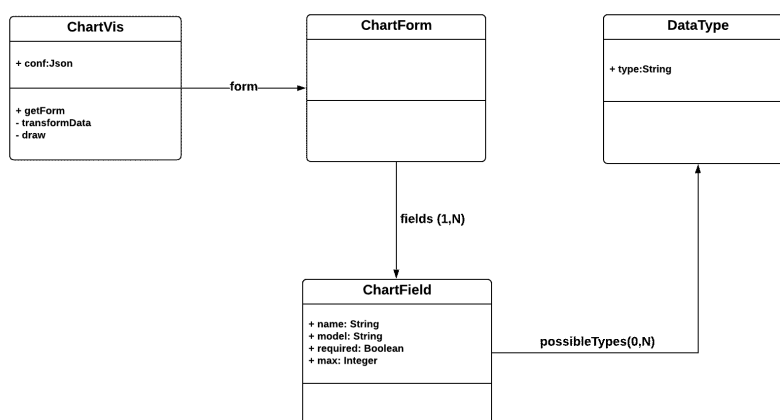


Figura 5.7: Clase ChartVis

La clase ChartVis debe tener un método 'getForm'. Este método devuelve un formulario para la creación del tipo de gráfico correspondiente. El formulario tiene: instrucción (instruction), la cual es una mera descripción para orientar a los usuarios con lo que se pretende que hagan al completarlo; un nombre (name), el cual también es descriptivo, es el título del formulario; y una lista de campos (fields). Los campos del formulario tienen una meta información, con la cual la aplicación sabe de que manera debe mostrar-

⁴<https://leafletjs.com/>

⁵<https://www.openstreetmap.org/>

lo y comportarse. Cada campo del formulario tiene los siguientes atributos: nombre (name), el cual es un nombre descriptivo; modelo (model), el cual permite hacerle referencia; máximo (max), indicando la cantidad máxima de columnas posibles dentro de este campo; y tipo (type), el cual es un arreglo con los tipos permitidos para dicho campo. El formulario, a medida que los usuarios lo completan, determina cómo se crea la configuración que el tipo de visualización necesita. El listing 5.1 es un ejemplo de formulario para el tipo de visualización Mapa.

```
1 const instruction = 'Selecciona una latitud, una longitud, y  
    los valores que desees incluir en el mapa'  
2 const name = 'Mapa'  
3 const fields = [  
4   {  
5     name: 'Latitud',  
6     type: [types.lat],  
7     model: 'latitude',  
8     required: true,  
9     max: 1  
10  },  
11  {  
12    name: 'Longitud',  
13    type: [types.lng],  
14    model: 'longitude',  
15    required: true,  
16    max: 1  
17  },  
18  {  
19    name: 'Otros',  
20    type: [],
```

```

21     model: 'others',
22     required: false,
23     max: null
24   }
25 ]

```

Listing 5.1: Formulario para crear visualización de tipo mapa

Si el valor en max es null, quiere decir que no tiene máximo de columnas para ser elegidas dentro del campo, si en cambio tiene un valor, se respeta el máximo indicado. Cuando el arreglo type está vacío, eso quiere decir que cualquier tipo puede ser asignado a ese campo del formulario. Los tipos disponibles a utilizar están definidos como constantes dentro de la aplicación y son los que aparecen en el listing 5.2.

```

1 const date = 'Fecha'
2 const number = 'Numerico'
3 const text = 'Texto'
4 const lat = 'Latitud'
5 const lng = 'Longitud'

```

Listing 5.2: Tipos de datos disponibles

Las clases que heredan de ChartVis, deben tener tres métodos: 'getForm', 'transformData', y 'draw'. En el párrafo anterior se hizo una descripción del método 'getForm'. El método 'draw', es el que sabe como dibujarse. Es el encargado de comunicarse con librerías externas, como por ejemplo d3, para la visualización del mismo. El método 'transformData', se llama antes de 'draw'. Este método crea un objeto para que el método 'draw' pueda dibujar sin preocuparse demasiado por como tratar la información. Esto lo hace manipulando la información contenida dentro del objeto NiceTable, aplicando

la configuración obtenida al completar el formulario.

Transformaciones

La biblioteca permite realizar distintas transformaciones para con los objetos de clase NiceTable. A continuación se listan las transformaciones implementadas.

- Filtrar columnas. Esto da lugar a eliminar a las que no sean de interés dejando una interfaz mas despejada[12]. De todas formas pueden ser recuperadas cuando sea necesario. La figura 5.8 muestra la interfaz que permite realizar esta acción.

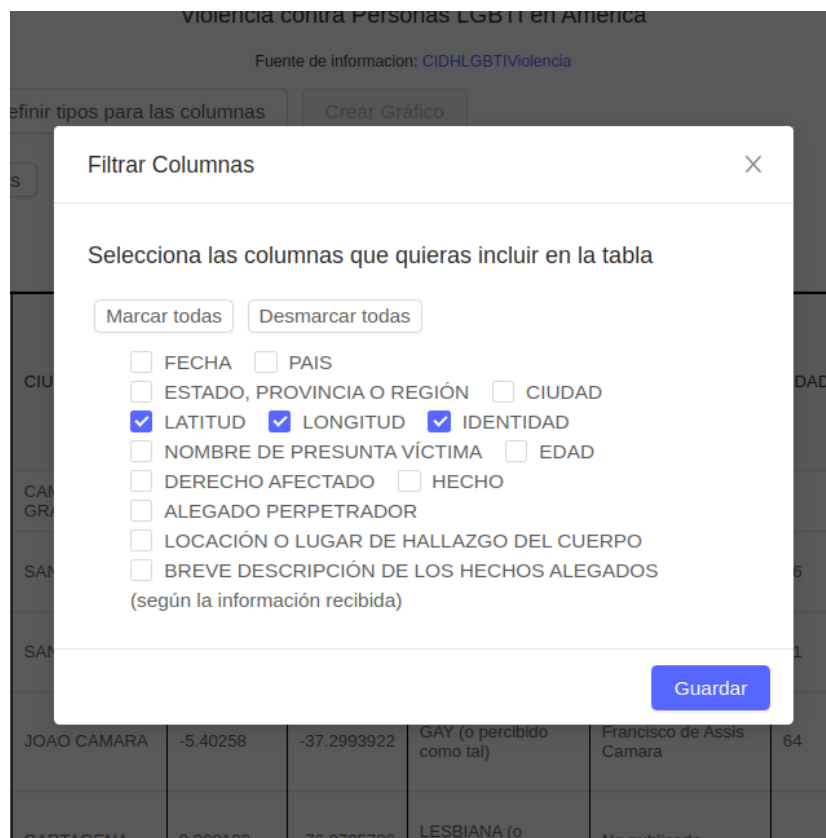


Figura 5.8: Filtrar columnas UI

- Renombrar columnas. En el caso de que las columnas originales tengan nombres poco descriptivos, los mismos pueden cambiarse para ser mas amigables con lxs usuarixs. La figura 5.9 muestra la interfaz que permite realizar esta acción.

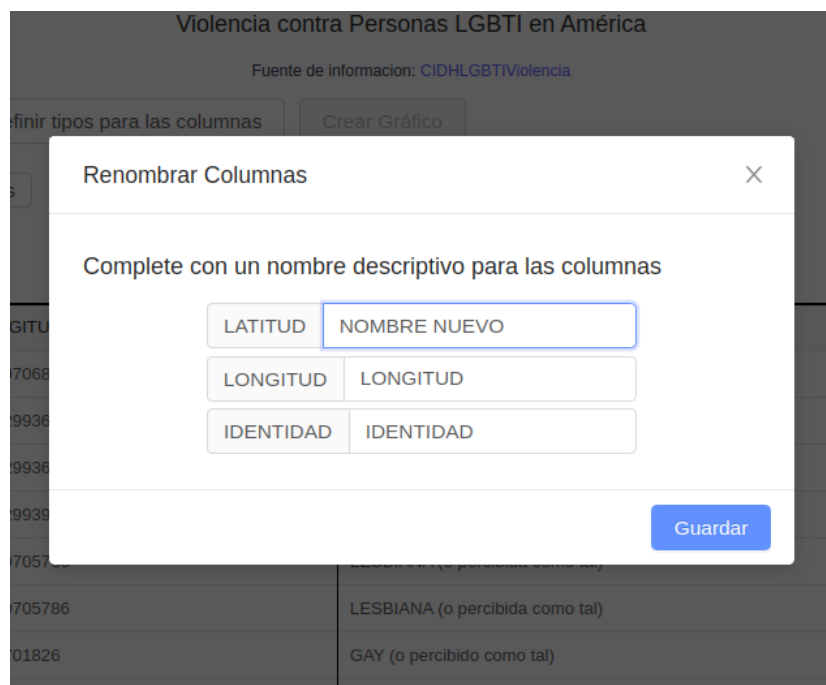


Figura 5.9: Renombrar columnas UI

- Cambiar el tipo correspondiente a cada columna. El tipo de las columnas permite buscar y ordenar en la tabla dentro de cada columna. También determina de que forma se utilizara la columna en los distintos gráficos que puedan generarse. La figura 5.10 muestra la interfaz que permite realizar esta acción.

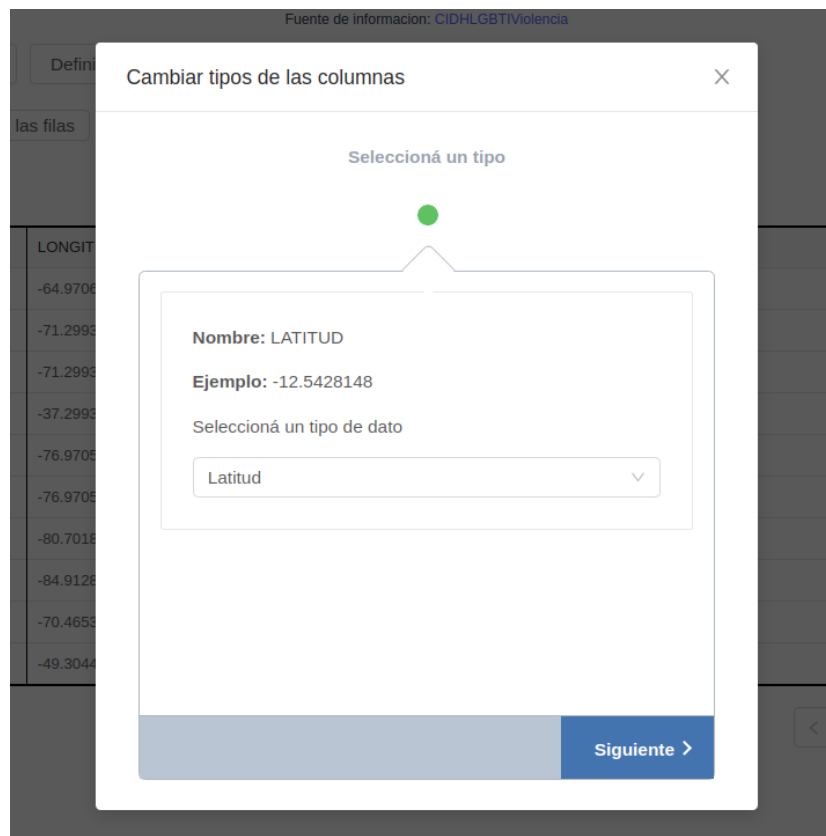


Figura 5.10: Cambiar tipo de datos de las columnas UI

La siguiente figura 5.11 corresponde a la clase Transformation, que representa las transformaciones mencionadas en el párrafo anterior. Las transformaciones hacen modificaciones en las columnas de las tablas. Cada transformación recibe un objeto NiceTable y lo devuelve con las modificaciones que le hayan sido aplicadas. La clase Transformation también podría extenderse agregándose nuevos tipos de transformaciones, siempre y cuando devuelvan un objeto de clase NiceTable.

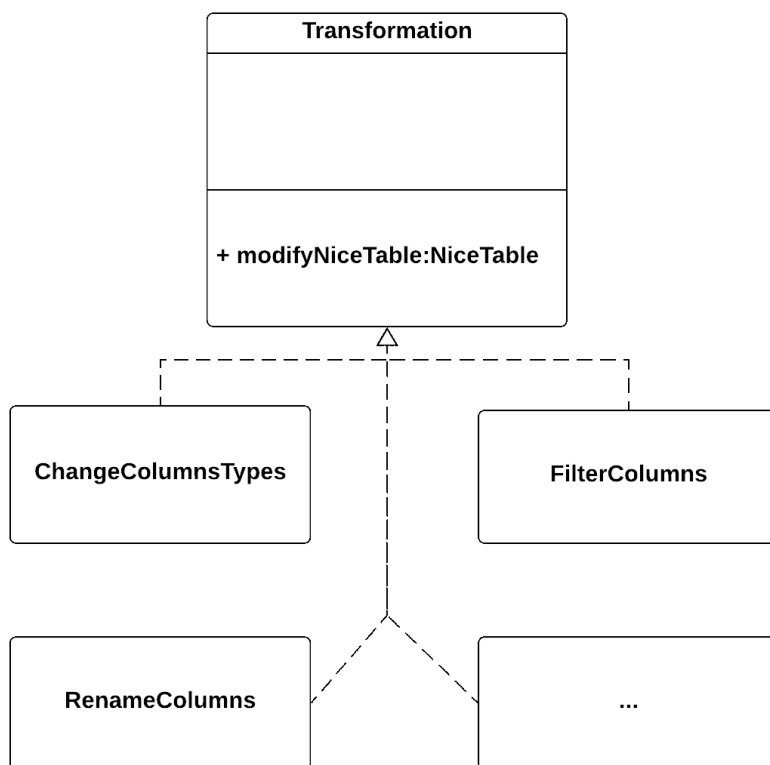


Figura 5.11: Clase Transformation

Persistencia

La herramienta permite que las interacciones con la herramienta sean persistentes. Esto quiere decir que información generada puede ser recuperada en cualquier momento. Para lograr dicha persistencia, la herramienta posee un backend. El backend fue desarrollado utilizando la tecnología Django REST Framework[21]. El mismo permite que las manipulaciones con las tablas y la eliminación y modificación de los gráficos sean persistentes. Para

alcanzar dicha persistencia, se crearon tres modelos Django (descritos en el listing 5.3), y por ende, tres tablas en la base de datos[22].

```
1 class TablePersistence(models.Model):
2     domain = models.CharField(max_length=100)
3     identificador = models.CharField(max_length=100)
4
5
6 class ColumnPersistence(models.Model):
7     table = models.ForeignKey(TablePersistence, on_delete=
8         models.CASCADE, related_name='columns')
9     index = models.CharField(max_length=100)
10    title = models.CharField(max_length=100)
11    column_type = models.CharField(max_length=100)
12    visible = models.BooleanField(default=True)
13
14 class ChartPersistence(models.Model):
15     table = models.ForeignKey(TablePersistence, on_delete=
16         models.CASCADE, related_name='charts')
17     chart_type = models.CharField(max_length=30)
18     conf = JSONField()
```

Listing 5.3: Modelos del backend

5.2. Instalación y uso

La librería esta creada con el framework javascript VueJS[23], por lo que, debemos incluir el mismo en cada sitio donde queramos instalar la misma. Se podrá instalar la librería obteniéndola del gestor de paquetes npm:

```
1 npm install infovis
```

Listing 5.4: Instalar TableVis

Luego de instalarla, debemos incluirla en nuestro sitio web. El siguiente fragmento de código muestra como instalarla en un proyecto vue existente:

```
1 import infovis from 'infovis'
2 import 'infovis/dist/infovis.css'
```

Listing 5.5: Importar TableVis en proyecto vue

En caso de no ser un proyecto vue, se incluye la librería de la siguiente manera:

```
1 <script src="https://unpkg.com/vue"></script>
2 <script src="path/to/infovis.umd.js"></script>
3 <link rel="stylesheet" href="path/to/infovis.css">
```

Listing 5.6: Importar TableVis sin vue

A continuación se muestra el fragmento de código que permite inicializar la librería para la tabla con id 'tableId', partiendo de un proyecto vue existente:

```
1 export default {
2   // mounted es un metodo que se acciona automaticamente en
3   // vue una vez que el sitio ha sido renderizado
4   mounted () {
5     // creamos la configuracon para inicializar la libreria
6     const conf = {
7       // El string tableId es el id de nuestra tabla html
8       extractor: infovis.extractors.tableIdExtractor.
9         getTable('tableId'),
9       // el elemento del dom con id 'tableId' se
10      // utilizara para insertar la tabla nueva
```

```

11     newTableInDiv: document.getElementById('tableId'),
12     // se utilizara el metodo replace para insertar la
13     // nueva tabla
14     tableInsertMethod: 'replace',
15     // elemento del dom donde se insertaran los graficos
16     // generados
17     chartsDiv: document.getElementById('charts')
18 }
19 // inicializamos la libreria indicando la configuracion
20 infovis.transformTable(conf)
21 }
22 }

```

Listing 5.7: Inicializar TableVis en proyecto vue

En caso de no ser un proyecto vue, se debe inicializar la librería de la siguiente forma:

```

1 <script>
2   new Vue({
3     // creamos la configuracon para inicializar la libreria
4     const conf = {
5       // El string tableId es el id de nuestra tabla html
6       extractor: infovis.extractors.tableIdExtractor.
7         getTable('tableId'),
8       // el elemento del dom con id 'tableId' se
9       // utilizara para insertar la tabla nueva
10      newTableInDiv: document.getElementById('tableId'),
11      // se utilizara el metodo replace para insertar la
12      // nueva tabla
13      tableInsertMethod: 'replace',
14      // elemento del dom donde se insertaran los graficos

```

```

14      // generados
15      chartsDiv: document.getElementById('charts')
16    }
17    // inicializamos la libreria indicando la configuracion
18    infovis.transformTable(conf)
19  })
20 </script>

```

Listing 5.8: Inicializar TableVis sin vue

Una vez inicializado el componente **TableVis**, se genera una nueva tabla partiendo de la tabla original. Por sobre la tabla, se visualizan una serie de botones correspondientes a las operaciones de transformación que pueden realizarse interactuando con la misma. Se permite seleccionar las columnas, dejando así solo las que sean pertinentes para los usuarios, eliminando las que no lo sean. También, se pueden renombrar las columnas de manera que sean más descriptivas. Esta nueva tabla agrega la funcionalidad de orden y búsqueda dependiendo el tipo de las columnas. Otra de las operaciones disponibles, es cambiar el tipo asociado a las columnas.

TableVis permite la creación de distintos tipos de gráficos seleccionando las filas y las columnas que se quieran incluir en los mismos. Dependiendo el gráfico seleccionado, habrá distintos requerimientos para la información seleccionada, para esto también es necesario que los tipos de las columnas estén bien asociados. La librería detecta automáticamente los tipos asociados a las mismas, pero en caso de no haberlo hecho de manera que los usuarios consideren correcta, podrán modificar dichas asociaciones. Hay un listado de gráficos disponibles para crear, cada uno de ellos con distintos tipos de campos requeridos para su visualización. Una vez creados los mismos, también

tendremos la posibilidad de eliminarlos.

La aplicación permite, al inicializarse, indicar mediante un campo si las operaciones deben realizarse o no con persistencia. Por defecto, si no se especifica dicha configuración, las operaciones serán persistentes. En caso de usarse la aplicación con persistencia, se guardan las interacciones en la base de datos pudiendo ser recuperadas en cualquier momento, de lo contrario, la aplicación funciona sin guardar las alteraciones que se realicen en las tablas y gráficos creados.

5.3. API

El backend brinda persistencia durante la utilización de **TableVis**. Está desarrollado con la tecnología Django REST Framework, utilizando la arquitectura REST. Su uso permite persistir las modificaciones en las tablas y la creación de los gráficos. Para la comunicación entre la interfaz y el backend, se desarrolló una API la cual dispone de una serie de urls que permiten hacer uso de las mencionadas funcionalidades. A continuación se describen las urls con las cuales **TableVis** interactúa para poder crear, eliminar y modificar los objetos correspondientes a su dominio en la base de datos[21].

Nombre	Obtener Tabla
URL	table/
Método	GET
Parámetros Url	domain=[String], identifier=[String]
Respuesta satisfactoria	<pre> 1 { 2 Response: Object , 3 Code: 200 4 }</pre>
Respuesta error	Código: 400

Cuadro 5.1: API: Obtener Tabla

Nombre	Crear Tabla
URL	table/
Método	POST
Parámetros Data	domain=[String], identificador=[String], co- lums=[Array]
Respuesta satisfactoria	<pre> 1 { 2 Response: Object , 3 Code: 201 4 }</pre>
Respuesta error	Código: 400

Cuadro 5.2: API: Crear tabla

Nombre	Listar Gráficos
URL	chart/
Método	GET
Parámetros Url	domain=[String], identificador=[String]
Respuesta satisfactoria	<pre> 1 { 2 Response: Array, 3 Code: 200 4 }</pre>
Respuesta error	Código: 400

Cuadro 5.3: API: Listar gráficos

Nombre	Crear Gráfico
URL	chart/
Método	POST
Parámetros Data	domain=[String], identifier=[String], conf=[Json]
Respuesta satisfactoria	<pre> 1 { 2 Response: Object , 3 Code: 201 4 }</pre>
Respuesta error	Código: 400

Cuadro 5.4: API: Crear gráfico

Nombre	Eliminar Gráfico
URL	chart_detail/(pk)
Método	DELETE
Parámetros URL	pk=[Integer]
Respuesta satisfactoria	<pre> 1 { 2 Code: 204 3 }</pre>
Respuesta error	Código: 404

Cuadro 5.5: API: Eliminar gráfico

5.4. Tecnologías

En esta sección se mencionan las principales tecnologías utilizadas durante el desarrollo de la herramienta. Se describe brevemente cada una de ellas.

Vuejs

Vue.js es un framework javascript. Es una librería fácil de usar e integrar con otros proyectos y bibliotecas existentes, así como también se usa para la creación de aplicaciones de una sola pagina (single-page-application o SPA). Programar con vue permite dividir la interfaz en pequeños componentes, los cuales a su vez pueden contener uno o muchos otros componentes. Dando la ventaja de hacer los mismos de manera que puedan ser reutilizables, tanto dentro del mismo proyecto como en otros. Los componentes combinan código HTML, javascript y css[23].

npm

Node Package Manager abreviada como npm⁶, es una herramienta usada por alrededor de 11.000.000 desarrolladores javascript alrededor del mundo. Es un sistema de gestión de paquetes que nos permite manejar las dependencias de una aplicación e instalar aplicaciones dentro de un repositorio.

d3

Data-Driven Documents o d3⁷ es una librería javascript utilizada para visualizar datos con HTML, SVG y CSS. Nos permite producir infogramas dinámicos e interactivos en navegadores web. A diferencia de otras librerías, permite tener el control completo sobre el resultado visual final.

⁶<https://www.npmjs.com/>

⁷<https://d3js.org/>

moment

Moment.js⁸ es una librería javascript que nos permite manipular de manera simple las fechas.

Django Rest Framework

Django Rest Framework es un proyecto colaborativo nacido del proyecto Django. Es una herramienta poderosa y flexible para crear web APIs. Lo que hace este framework es simplificar el proceso de lectura de la información que tenemos almacenada en nuestra base de datos a través de la descripción provista definida en los modelos. Es un framework muy fácil de configurar y utilizar[22][21].

Postgres

Postgres es una base de datos relacional que usa y extiende el lenguaje SQL combinado con otras funciones que lo vuelve mas seguro y escalable[24].

Git Version Control

Es un sistema que guarda el rastro de todos los cambios en los archivos de código del proyecto permitiéndonos acceder a cualquier versión en cualquier momento. También brinda mucha flexibilidad a la hora de trabajar en equipo, que si bien no es el caso de este desarrollo por el momento, permite dejar el código disponible para futuras actualizaciones en colaboración de otros desarrolladores[25].

⁸<https://momentjs.com/>

Capítulo 6

Evaluación

En este capítulo se realiza una evaluación de la herramienta desarrollada desde el punto de vista del usuario final. Teniendo ya una versión estable de la herramienta, se planificó una prueba formal para evaluar la interacción de los usuarios con la misma. Se propuso un experimento para el cual participaron 27 personas. En dicho experimento se proponen objetivos a cumplir interactuando con una tabla a la cual se le agregaron las funcionalidades provistas por **TableVis**. Para realizar la evaluación, se creó una página web. Dentro de la misma hay una breve introducción sobre cómo se estructura la evaluación e instrucciones con los pasos correspondientes para realizarla. La misma incluye un video demostrativo sobre cómo interactuar con la herramienta. De los participantes primero se solicita información personal y una vez finalizadas las tareas propuestas se pide un feedback. La siguiente figura 6.1 corresponde a una captura de pantalla de la página web que se creó para realizar la evaluación.

Evaluación

Gracias por participar en esta experiencia. La misma tiene como objetivo realizar una evaluación sobre una herramienta llamada **TableVis** realizada en el contexto de una tesina para la Facultad de Informática, UNLP. **TableVis** se trata de una librería que puede ser incluida por programadores en sus aplicaciones web y así brindar a los usuarios ciertas funcionalidades a la hora de interactuar con tablas de información.

Esta experiencia se divide en tres partes:

- Primero una encuesta inicial con preguntas sobre información demográfica.
- Luego se proponen una serie de tareas para interactuar con la herramienta desarrollada.
- Por último, una encuesta final evaluando dicha interacción.

1. Encuesta inicial

Por favor, conteste la siguiente encuesta: <https://forms.gle/v67KbaxZve4HdshaA>

2. Interacción con TableVis

El siguiente video, muestra unos ejemplos de interacción con TableVis, se recomienda mirarlo antes de empezar para tener una orientación sobre su uso.



A continuación, se describen las tareas propuestas a realizar con una tabla de ejemplo. Primero leelas, al final vas a encontrar el enlace donde realizarías.

2.1 Filtrar: Quedarse sólo con las siguientes columnas:

- LATITUD
- LONGITUD
- IDENTIDAD

Figura 6.1: Pagina web evaluación

La evaluación se estructuró en tres etapas:

1. Encuesta Inicial
2. Tareas propuestas
3. Encuesta Final

Las siguientes secciones describen en detalle sobre cada una de las etapas mencionadas.

6.1. Encuesta inicial

Antes de realizar las tareas propuestas y evaluar la interacción de los participantes con la herramienta, se le pide a cada uno que conteste una breve encuesta. Esta encuesta inicial es acerca de información demográfica. La encuesta fue realizada utilizando formularios de Google, ya que brindan muchas facilidades para la recolección de resultados y su posterior análisis. La encuesta inicial tiene dos secciones: la primera sobre información demográfica, y la segunda sobre experiencia con la tecnología.

6.1.1. Información demográfica

A continuación se muestran los resultados de cada una de las preguntas realizadas en la primera sección de la encuesta inicial.

Nacionalidad

Como se ve en la figura 6.2, participaron de la evaluación un 96,3% personas con nacionalidad argentina y 3,7% con nacionalidad española.



Figura 6.2: Encuesta Inicial - Nacionalidad

Género

Como muestra la figura 6.3, de las personas que participaron en la evaluación, el 48,1 % es de género femenino, y el 51,9 % de género masculino.

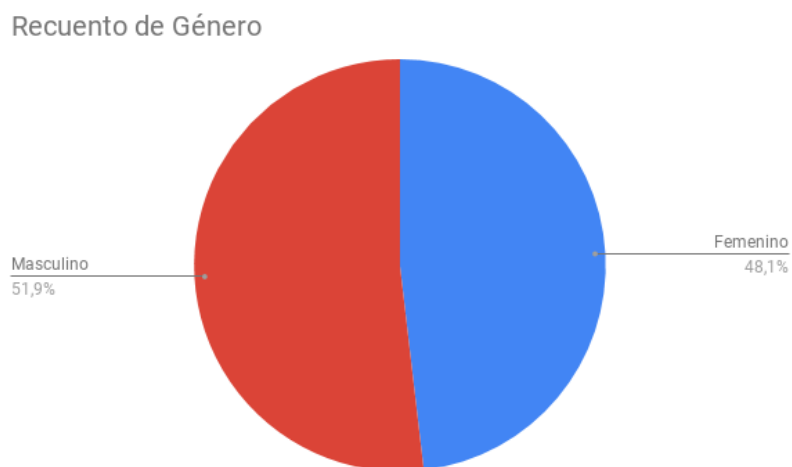


Figura 6.3: Encuesta Inicial - Género

Encuesta Inicial - Nivel de educación

De todas las personas que participaron en la evaluación, como podemos observar en la figura 6.4, el 59,3 % recibió educación terciaria/universitaria, el 29,6 % educación secundaria, y el 11,1 % educación de postgrado.

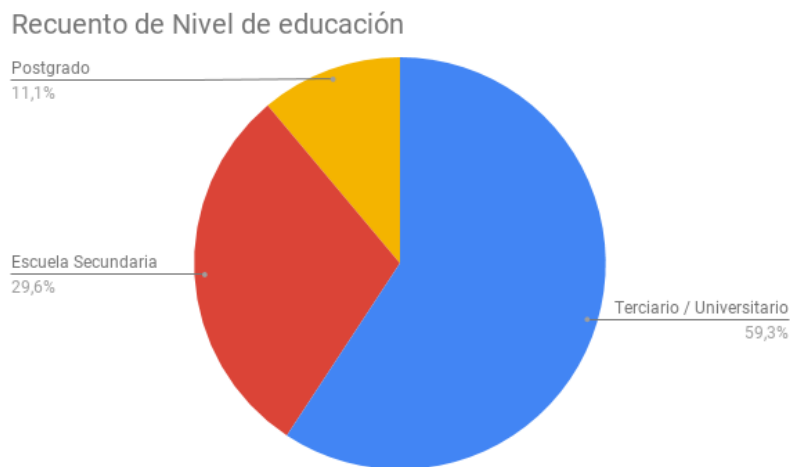


Figura 6.4: Nivel de educación

Rubro

En esta evaluación participaron personas de distintos rubros. La mayor cantidad de participantes, un 25,9 %, se relaciona con la rama del arte. En segundo lugar, un 25 % se relaciona con la rama de la informática. En tercer lugar está la rama de las ciencias sociales, con un 22,2 % de lxs participantxs. Luego, están las ramas de la aviación; marketing; arquitectura; políticas y comerciales; y economía, cada una con un 3,7 % de participantes. En la figura 6.5 podemos ver representados dichos porcentajes.



Figura 6.5: Encuesta Inicial - Rubro

Estado Profesional

Por último, en la sección de información demográfica, hay una pregunta acerca del estado profesional. El 81,5 % de lxs participantxs son trabajadorxs, el 11,1 % son estudiantxs, y el 7,4 % se encuentra buscando trabajo. Podemos ver estos porcentajes representados en la figura 6.6.

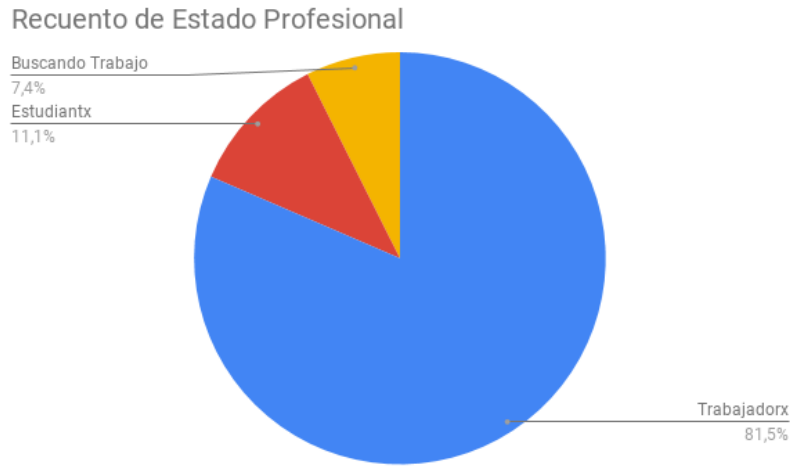


Figura 6.6: Encuesta Inicial - Estado Profesional

6.1.2. Relación con la tecnología

A continuación se describen las preguntas realizadas en la segunda sección de la encuesta inicial. Esta sección realiza preguntas sobre la relación que cada participante tiene con la tecnología.

¿Con qué frecuencia utilizas internet?

Como ilustra la figura 6.8, todxs lxs participantes involucradxs en esta evaluación usan internet a diario.

Recuento de Con qué frecuencia utilizas internet?

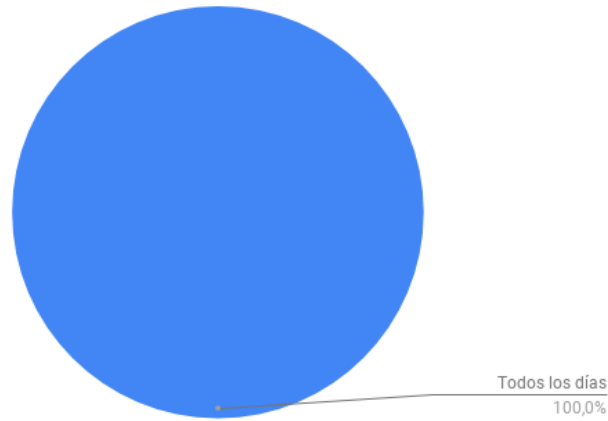


Figura 6.7: Encuesta Inicial - Frecuencia del uso de internet

¿Con qué frecuencia utilizas internet sin contar el uso de redes sociales?

Internet puede utilizarse con múltiples propósitos. Algunas personas, si no trabajan o estudian con la computadora, puede que gran parte de su uso de internet sea en las redes sociales. De lxs participantes, 81,5 % usan internet a diario sin contar el uso de las redes sociales; 14,8 % lo hacen algunas veces a la semana; y el 3,7 % nunca lo hace. La figura 6.8 muestra dichos resultados.

Recuento de Con qué frecuencia utilizas internet sin contar el uso de redes sociales?

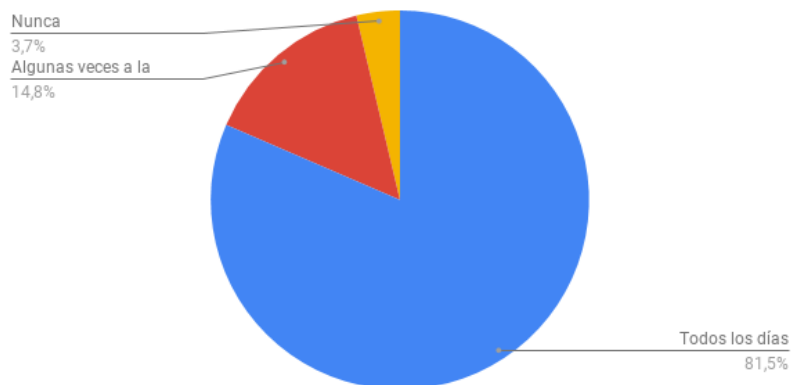


Figura 6.8: Encuesta Inicial - Frecuencia del uso de internet sin contar redes sociales

¿Con cuánta frecuencia utilizas el navegador web?

Como puede observarse en la figura 6.9, de todxs lxs participantxs, 85,2 % hace uso del navegador web a diario, 11,1 % lo hace algunas veces a la semana, y 3,7 % lo hace raramente.

Recuento de Con cuánta frecuencia utilizas el navegador web?

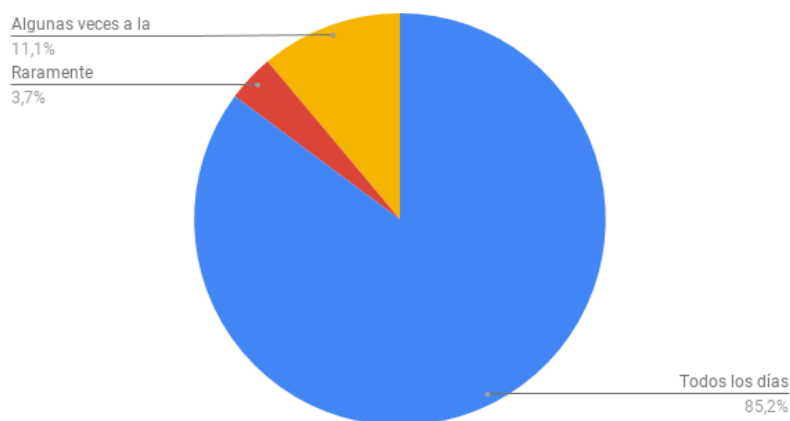


Figura 6.9: Encuesta Inicial - Frecuencia del uso del navegador

¿Con qué frecuencia utilizas hojas de cálculo (ej. Excel)?

De todos los participantes, el 33,3 % usa hojas de cálculo raramente, el 25,9 % algunas veces al mes, el 22,2 % algunas veces a la semana, el 14,8 % todos los días y el 3,7 % nunca lo hace. Dichos porcentajes se observan en la figura 6.10.



Figura 6.10: Encuesta Inicial - Frecuencia del uso de hojas de calculo

¿Alguna vez creaste gráficos a través de una hoja de cálculo?

De las 27 personas que participaron, como puede verse en la figura 6.11, el 63 % ha creado gráficos alguna vez a través de una hoja de calculo, y el 37 % no lo ha hecho nunca.

Recuento de Alguna vez creaste gráficos a través de una hoja de cálculo?

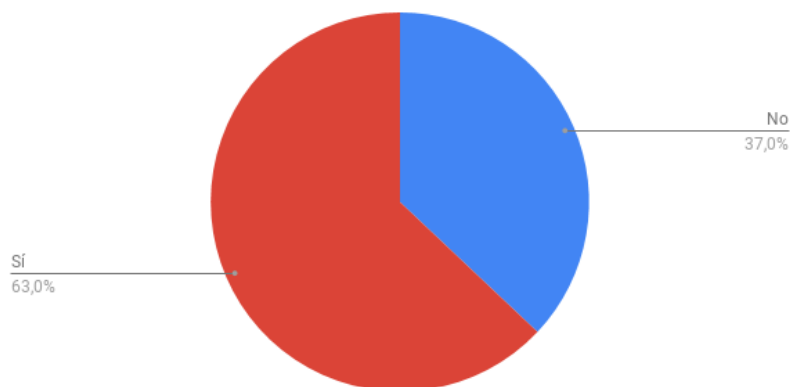


Figura 6.11: Encuesta Inicial - Experiencia creando gráficos a través de hojas de cálculo

¿Con cuánta frecuencia utilizas gráficos a través de hojas de cálculo (ej. Excel)?

Como muestra la figura 6.12, el 48,1 % nunca utiliza gráficos a través de hojas de calculo, el 33,3 % lo hace raramente, el 11,1 % lo hace algunas veces al mes, el 3,7 % lo hace algunas veces a la semana y el 3,7 % lo hace todos los días.

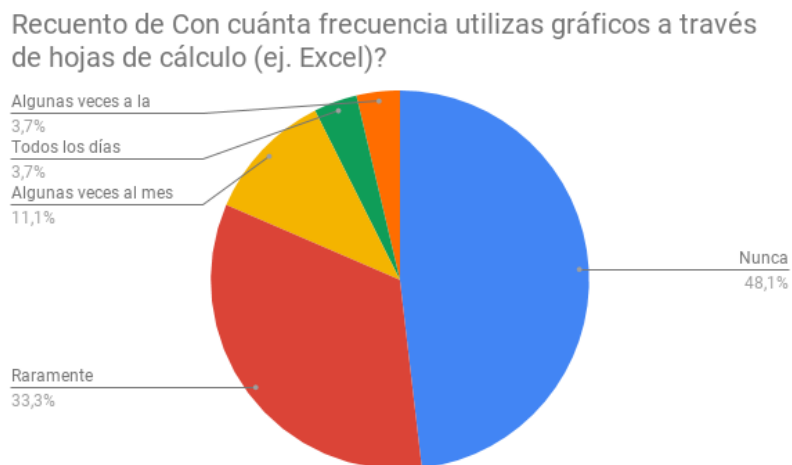


Figura 6.12: Encuesta Inicial - Experiencia creando gráficos a través de hojas de cálculo

¿Cuál es tu experiencia con herramientas que impliquen manipulación de tablas y gráficos?

De todxs lxs participantxs, el 55,6 % tiene una experiencia básica utilizando herramientas de manipulación de tablas y gráficos, el 22,2 % indicaron que su experiencia es nula, el 18, 5 % tienen experiencia media, y el 3,7 % tiene experiencia avanzada. Dichos porcentajes pueden verse en la figura 6.13.



Figura 6.13: Encuesta Inicial - Experiencia con herramientas de manipulación de tablas y gráficos

6.2. Tareas propuestas

Después de haber contestado la encuesta inicial, se pide a los participantes que realicen unas tareas específicas. Las mismas consisten en interactuar con un ejemplo demostración de la herramienta desarrollada en funcionamiento. Se instanció la herramienta en una tabla que registra violencia contra personas LGBTI en américa. Las tareas propuestas son las siguientes:

1. Filtrar: Quedarse sólo con las siguientes columnas: LATITUD, LONGITUD, IDENTIDAD, DERECHO AFECTADO, HECHO, LOCACIÓN O LUGAR DE HALLAZGO DEL CUERPO.
2. Renombrar: Cambie el nombre a la columna “LOCACIÓN O LUGAR

DE HALLAZGO DEL CUERPO” por el nombre “LUGAR DEL HECHO”.

3. Crear gráfico: Con la ayuda del botón Marcar todas las filas, seleccione todos los registros de la tabla y obtenga cuál es el porcentaje correspondientes a cada IDENTIDAD utilizando un gráfico de torta.
4. Crear gráfico: Con la ayuda del botón Marcar todas las filas, seleccione todos los registros y obtenga el HECHO que ocurrió más cantidad de veces utilizando un gráfico de barras.
5. Crear gráfico: Busque los registros de personas que poseen para el campo IDENTIDAD el valor MUJER TRANS (o percibida como tal). Seleccione los primeros 10 resultados. Con dichos resultados, cree un mapa incluyendo las columnas LATITUD, LONGITUD, HECHO y DERECHO AFECTADO. Cree una categoría dentro del mapa para la columna DERECHO AFECTADO.

Se pide a cada participante que de empezar a realizar las tareas anote la hora. Al finalizar se les pide que calculen cuanto tiempo les llevó realizarlas.

6.3. Encuesta Final

La encuesta final, tiene tres secciones. La primera sección indaga sobre si pudieron completar o no las tareas, por qué y cuanto tiempo les llevo completarlas. La segunda, siguiendo el formulario SUS, tiene preguntas respecto a la usabilidad de la herramienta. La tercera, hace preguntas sobre ventajas, desventajas y mejoras posibles en la herramienta.

6.3.1. Respecto a las tareas propuestas

¿Pudo completar todas las tareas propuestas?

Como se puede ver en la figura 6.14, el 96,3 % de lxs participantes pudieron completar las tareas, y el 3,7 %. En la siguiente pregunta se indica el motivo.



Figura 6.14: Encuesta Final - ¿Pudieron completar todas las tareas propuestas?

En caso de no haberlas podido completar. ¿Cuál fue el motivo?

Hubo sólo un participante que no pudo completar las tareas propuestas. La complicación fue con la última tarea propuesta. El motivo fue no entender cómo filtrar la información dentro de una columna. Más adelante se menciona nuevamente dicho inconveniente, dando mas detalles sobre el mismo.

¿Cuánto tiempo te llevó completarlas?

En la figura 6.15 se ve que la mayoría de lxs participantxs tardó menos de 10 minutos en realizar las tareas. Más específicamente, 21 de ellxs tardaron menos de 10 minutos (77,8 %), 4 tardaron 10 minutos (14,8 %) y 2 tardaron mas de 10 minutos (7,4 %).

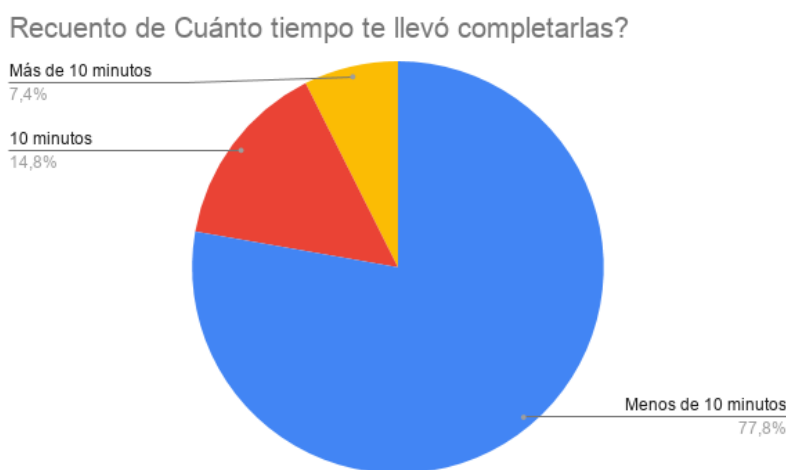


Figura 6.15: Encuesta Final - Tiempo en realizar las tareas

6.3.2. Usabilidad

En esta sección del formulario, se evalúa la usabilidad de la herramienta utilizando el formulario estandarizado SUS (System Usability Scale). Este método de medición consiste en 10 preguntas con opciones de repuestas numéricas. Las posibles respuestas varían del 1 al 5, siendo la opción 1 **Totalmente en desacuerdo** y la opción 5 **Totalmente de acuerdo**. Los puntajes del formulario SUS varían entre 1 y 100. El puntaje 68 es conside-

rado el puntaje promedio[26]. Las preguntas traducidas correspondientes al formulario son las siguientes:

1. Me gustaría usar la herramienta frecuentemente
2. Encuentro esta herramienta innecesariamente compleja
3. Pienso que esta herramienta es fácil de usar
4. Pienso que necesitaría ayuda para poder usar esta herramienta
5. Encuentro muchas funciones de esta herramienta bien integradas
6. Pienso que hay demasiada inconsistencia en esta herramienta
7. Pienso que la mayoría de la gente aprendería a usar esta herramienta rápidamente
8. Encuentro la herramienta muy difícil de usar
9. Me siento segurx usando esta herramienta
10. Tuve que aprender muchas cosas antes de poder utilizar esta herramienta

Para calcular los resultados, se suman los resultados obtenidos considerando lo siguiente: las preguntas impares toman el valor asignado por lxs usuarixs y se le resta 1; las preguntas pares le restan el valor asignado por lxs usuarixs al numero 5. Una vez obtenido el numero final, se lo multiplica por 2,5[26]. A continuación, la figura 6.16 muestra el recuento de resultados del formulario SUS.

SUS Calculation		1: Strongly disagree					5: Strongly agree					
Participant / Question	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	SUS Score	
p1	5	1	4	2	5	1	2	5	5	1	77,5	
p2	3	2	4	2	3	1	2	4	5	1	67,5	
p3	2	1	5	1	3	2	1	5	5	1	65,0	
p4	5	2	3	2	4	1	3	3	4	1	75,0	
p5	5	1	5	2	5	1	1	5	5	2	75,0	
p6	4	1	5	2	5	1	1	5	5	1	75	
p7	5	1	5	1	5	1	1	5	3	1	75	
p8	5	1	4	1	4	1	2	4	5	1	80	
p9	3	1	4	2	4	1	2	3	5	1	75	
p10	4	1	5	1	5	1	1	5	5	1	77,5	
p11	1	2	5	1	5	1	1	5	5	1	67,5	
p12	4	2	4	1	5	1	2	5	5	1	75,0	
p13	5	2	4	2	5	2	2	5	5	3	67,5	
p14	3	1	5	1	5	1	1	4	4	2	72,5	
p15	2	2	4	1	3	1	2	2	4	1	70,0	
p16	5	1	5	1	5	1	1	4	5	1	82,5	
p17	2	3	2	4	3	1	4	3	3	1	55,0	
p18	4	1	4	2	5	1	2	4	4	1	75,0	
p19	5	4	4	3	4	2	2	5	5	2	60,0	
p20	4	1	1	5	2	5	5	1	5	1	60,0	
p21	5	1	5	1	5	1	1	1	5	1	90,0	
p22	4	1	4	3	3	1	2	3	4	2	67,5	
p23	5	1	5	1	5	1	1	5	5	1	80,0	
p24	3	1	4	2	5	1	2	4	5	3	70,0	
p25	2	2	3	1	4	2	3	1	2	4	60,0	
p26	5	1	4	1	5	1	2	4	5	1	82,5	
p27	5	1	5	1	5	1	1	5	5	1	80,0	
Average:											72,5	

Figura 6.16: Formulario SUS

6.3.3. Feedback

Esta es la última sección de la segunda encuesta. Se hacen preguntas a desarrollar sobre opiniones de la herramienta. A continuación se encuentran las preguntas realizadas y sus respuestas.

Podrías mencionar ejemplos de situaciones en las que podría usarse la herramienta?

Entre las respuestas de lxs participantes, surgieron diversas situaciones en las que pensaron podría ser útil el uso de la herramienta: para informar sobre estadísticas con respecto a la violencia de genero, femicidios, etc; para inseguridad y consumo; para mantener estadísticas sobre violencia contra personas LGBTI y poder generar alertas, elaborar estrategias de prevención, obtener lugares más afectados para proveer más educación, dar conocimiento y generar consciencia de la situación actual, etc; para visualización de bases de datos y/o indexación de datos numéricos de cualquier orden; para un manejo de los datos relacionados a cuestiones sociales de una manera practica, pragmática y versátil; análisis demográfico de desarrolladores de software; para backoffice de cualquier aplicación; para datos demográficos y censos; para realizar el análisis y fundamentar un proyecto de arquitectura y/o urbanística, establecer los recorridos de sitios más frecuentados, más peligrosos, más agradables etc; para investigación, análisis de mercado, ventas; demografía del voto, natalidad, drogas, enfermedades, femicidios; para vincular datos tanto sociológicos como prácticos, por ejemplo para vincular físicamente necesidades de tipo comercial, facilitar el entendimiento de la coyuntura económica y social de una zona en particular con respecto a tipos de consumo o bien zonificar la competencia en un rubro específico.

Tuviste algún problema con la herramienta? Cuál?

La mayoría de lxs participantes no tuvo problema en realizar las actividades. Hubo una minoría que si lo tuvo, y la misma coincidió en que la

lupa para filtrar por columnas no es intuitivamente fácil de localizar. A estxs participantes les costó encontrar la lupa para poder filtrar dentro de una columna. Incluso hubo una participante que no supo como hacerlo.

Menciona 3 cosas positivas de la herramienta

Dentro de las cosas positivas de la herramienta, se mencionan distintos aspectos. Les pareció fácil de usar, simple, útil e informativa. Resulta entendible para crear gráficos de manera rápida sin necesidad de tener conocimientos específicos. Permite interpretar muchos datos a la vez de manera sencilla, cosa que sin esta herramienta seria algo complejo de relacionar. Brinda la posibilidad de generar material gráfico con mucha rapidez, también permite comparar fácilmente los gráficos. Las funcionalidades están bien identificadas/etiquetadas. Nuclea los tipos de gráficos más usados, haciendo más fácil de distinguir cuál necesito o busco. Filtra y ordena fácilmente. Es veloz para la generación de respuestas. Las posibilidades y marcos de aplicación son muy amplios, lo cual la convierte en una herramienta interesante por sus posibles funcionalidades.

Menciona 3 cosas negativas de la herramienta

Como se menciona anteriormente, una desventaja en la que coincidieron algunxs de lxs participantes, es que no fue fácil de encontrar la lupa para realizar búsquedas dentro de las columnas de la tabla. A un participante le pareció difícil entender a primera vista que información iba a mostrar cada gráfico y que datos necesita sin haberlo utilizado antes. Otra participante indicó que le gustaría ingresar su propia información y crear nuevas columnas,

hacerlo aun mas estándar. Dos participantes coincidieron en que seria bueno elegir otra forma de visualizar los gráficos, por ejemplo poder crear solapas para agruparlos de distintas formas. Otra desventaja que indicaron es que no existe posibilidad de interactuar entre los gráficos ya creados. Sugirieron mejorar el estilo de la aplicación, que sea adaptable para dispositivos móviles.

Que cosas mejorarías de la herramienta o cuales propondrías cambiar?

Dentro de los comentarios de lxs participantes, surgieron varias propuestas de cosas a mejorar. A raíz del problema para encontrar la lupa, surge la posibilidad de cambiar el modo de búsqueda dentro de la tabla. Propusieron ubicarlo fuera de la tabla, incluso como un filtro general. Otra mejora podría ser agregar una breve descripción de objetivos e instrucciones para cada gráfico para poder entender su utilidad. La posibilidad de descargar los gráficos. Mayor vinculación entre los distintos tipos de gráficos. La posibilidad de agregar columnas, que no sean limitadas, por si se necesitan incluir más.

6.4. Conclusiones del capítulo

Gracias a esta evaluación, obteniendo la respuesta de las personas involucradas en la misma, pueden sacarse algunas conclusiones y sugerencias. Por empezar, en términos generales resulta fácil y rápida de utilizar, aun así siendo los participantes de diferentes rubros y edades. Como único inconveniente, hay un problema con el filtrado de información dentro de las

columnas, es visualmente difícil de encontrar para poder utilizarlo. Como sugerencias, surgen ideas de cambios y funcionalidades nuevas:

- Búsqueda general. Visualmente fuera de la tabla.
- Posibilidad de descargar gráficos.
- Posibilidad de cargar información.
- Posibilidad de agregar columnas nuevas.
- Posibilidad de relacionar gráficos entre sí.
- Visualizar los gráficos de otra manera. Posibilidad de agruparlos en solapas por ejemplo.

Estas sugerencias y problemáticas son de gran utilidad para ser tenidas en cuenta a la hora de realizar trabajos futuros. Otro aspecto a destacar es que si bien la herramienta fue desarrollada con objetivo de ser utilizada en aplicaciones relacionadas con la ciencia ciudadana, a lxs participantes se les ocurrieron muchos otros ejemplos sobre cómo podría ser útil su uso.

Capítulo 7

Conclusiones y trabajo futuro

7.1. Conclusiones

Una de las intenciones con el trabajo realizado en esta tesina es compartir conocimiento. No todos los desarrolladores pueden implementar cada funcionalidad que podría ser útil para los usuarios finales. La interacción con tecnologías que se desconocen lleva de mucho esfuerzo y tiempo, lo cual es muy valioso y la mayoría de las veces hace falta. A su vez, eso que algunos desarrolladores pueden desconocer, hay muchos otros que tienen conocimiento sobre cómo implementarlo[4]. El desarrollo de **TableVis** implicó investigar y aprender sobre la librería d3, cuya curva de aprendizaje es empinada. Con la publicación y libre distribución de dicha librería se le ahorra a otras personas que tengan que aprender a implementar los tipos de visualizaciones que están disponibles dentro de la misma[27].

Al dejar a disposición de los usuarios finales un grupo de operaciones disponibles, queda bajo su responsabilidad la creación de visualizaciones. Esto

hace que la interacción con los sitios sea dinámica. Los mismos no dependen de que sus administradorxs tengan que actualizarlo implementando nuevos tipos de visualizaciones, dicha implementación está provista por **TableVis** y visualizarlas queda en responsabilidad de lxs usuarixs[4].

7.2. Trabajo Futuro

Uno de los trabajos que se proponen a futuro es extender la herramienta desarrollada en distintos aspectos. **TableVis** implementa una cantidad limitada de visualizaciones, operaciones y extractores, pero fue desarrollada de manera tal que pueda ser extendida por cualquier persona con conocimientos en programación. Podrían agregarse nuevos tipos de gráficos, de operaciones de manipulación para con la tabla y distintos extractores. Por el momento, como se mencionó en el capítulo *Desarrollo Propuesto*, se implementó un tipo de extractor. El mismo, a partir del atributo id extrae las columnas y filas que hay en un elemento html con etiqueta table. Podría implementarse un nuevo tipo que lo haga a partir del atributo class¹, o podrían implementarse extractores que obtengan información de tablas estructuradas con etiquetas de otro tipo, como por ejemplo con la etiqueta div². También es posible extender los tipos de operaciones. Por ejemplo, podría agregarse una nueva operación que permita cambiar el orden en el que aparecen las columnas, o una que posibilite la división de las mismas. Las operaciones, para ser compatibles con **TableVis**, deben devolver siempre un objeto de clase NiceTable.

Existen múltiples tipos de visualizaciones que podrían agregarse a **Table-**

¹Clase CSS que se aplica a los estilos del elemento

²Estructura HTML para definir una división o sección

Vis. Para hacerlo, deben respetarse los métodos obligatorios que se describen en el capítulo de *Desarrollo Propuesto*.

A raíz de la evaluación desarrollada en el capítulo anterior, surgieron otras ideas para trabajos futuros gracias a los comentarios de lxs participantes. Se propone optimizar la búsqueda dentro de la tabla, una búsqueda general y más poderosa visualmente. Agregar la funcionalidad de descargar los gráficos realizadas y las tablas con las modificaciones realizadas. La posibilidad de cargar información. También se propone una nueva forma de visualizar los gráficos, podrían agruparse según distintos aspectos para una rápida futura recuperación, los mismos podrían tener títulos y comentarios.

Para seguir extendiendo el trabajo desarrollado en esta tesina, además de los trabajos propuestos en el párrafo anterior, se propone la implementación de una extensión³ para el navegador web. Se plantea que la misma permita agregar la funcionalidad provista por **TableVis** en aplicaciones web, sin la necesidad de que lxs programadorxs que las administran tengan que encargarse de la instalación de la misma.

³Pequeños programas que se instalan dentro del navegador

Bibliografía

- [1] T. Munzner, “Visualization analysis and design, ser. ak peters visualization series. ak peters, 2014.”
- [2] S. Finkelievich and C. Fischnaller, “Ciencia ciudadana en la sociedad de la información: nuevas tendencias a nivel mundial,” 2014.
- [3] G. Bosetti, S. Firmenich, M. Winckler, G. Rossi, U. C. Fandos, and E. Egyed-Zsigmond, “An end-user pipeline for scraping and visualizing semi-structured data over the web,” in *International Conference on Web Engineering*, pp. 223–237, Springer, 2019.
- [4] M. Toomim, S. M. Drucker, M. Dontcheva, A. Rahimi, B. Thomson, and J. A. Landay, “Attaching ui enhancements to websites with end users,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1859–1868, ACM, 2009.
- [5] J. Butler, *El género en disputa: el feminismo y la subversión de la identidad*. Paidós, 2007.

- [6] L. Dali, D. Rusu, and D. Mladenicić, “Enhanced web page content visualization with firefox,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 718–721, Springer, 2009.
- [7] M. Khan and S. S. Khan, “Data and information visualization methods, and interactive mechanisms: A survey,” *International Journal of Computer Applications*, vol. 34, no. 1, pp. 1–14, 2011.
- [8] M. Card, *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.
- [9] M. D. T. Ponjuán, “La visualización de la información en el entorno de la ciencia de la información,” 2010.
- [10] G. N. Dapozo, E. Irrazábal, *et al.*, “Xx workshop de investigadores en ciencias de la computación-wicc 2018: libro de actas,” 2018.
- [11] S. R. Martig, S. M. Castro, and S. Di Luca, “Interacción en la visualización de información,” in *IV Workshop de Investigadores en Ciencias de la Computación*, 2002.
- [12] B. Shneiderman, “The eyes have it: A task by data type taxonomy for information visualizations,” in *The craft of information visualization*, pp. 364–371, Elsevier, 2003.
- [13] M. Tory and T. Moller, “Rethinking visualization: A high-level taxonomy,” in *IEEE Symposium on Information Visualization*, pp. 151–158, IEEE, 2004.

- [14] C. N. Knaflitz, *Storytelling with data: A data visualization guide for business professionals*. John Wiley & Sons, 2015.
- [15] A. Fernández and D. Torres, “Ciencia ciudadana:¿ qué es?,” *Didáctica y TIC. Blog de la Comunidad Virtual de Práctica Docentes en Línea*, 2016.
- [16] D. Torres, F. Correa, A. Marisi, E. Claramunt, V. Cepeda, L. Lus, J. Ramírez, S. Pravisani, A. Díaz, M. Fressoli, *et al.*, “Cientópolis: motorizando la ciencia ciudadana,” in *IV Congreso Internacional Científico y Tecnológico-CONCYT 2017*, 2017.
- [17] M. Marasoiu, D. Nauck, and A. F. Blackwell, “Cuscus: An end user programming tool for data visualisation,” in *International Symposium on End User Development*, pp. 115–131, Springer, 2019.
- [18] D. Q. Nguyen and H. Schumann, “Visualization to support augmented web browsing,” in *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 01*, pp. 535–541, IEEE Computer Society, 2013.
- [19] F. B. Viegas, M. Wattenberg, F. Van Ham, J. Kriss, and M. McKeon, “Manyeyes: a site for visualization at internet scale,” *IEEE transactions on visualization and computer graphics*, vol. 13, no. 6, pp. 1121–1128, 2007.

- [20] K. Pantazos and S. Lauesen, “Constructing visualizations with in-fovis tools-an evaluation from a user perspective,” *GRAPP/IVAPP*, vol. 2012, pp. 731–736, 2012.
- [21] M. Masse, *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*. .O’Reilly Media, Inc.”, 2011.
- [22] A. Holovaty and J. Kaplan-Moss, *The definitive guide to Django: Web development done right*. Apress, 2009.
- [23] E. You, “Vuejs: The progressive javascript framework,” *Dostupné z: <https://vuejs.org>*.
- [24] E. Geschwinde and H.-J. Schönig, *PostgreSQL developer’s handbook*. Sams Publishing, 2002.
- [25] R. Somasundaram, *Git: Version control for everyone*. Packt Publishing Ltd, 2013.
- [26] B. Klug, “An overview of the system usability scale in library website and system usability testing,” *Weave: Journal of Library User Experience*, vol. 1, no. 6, 2017.
- [27] M. Mărăsoiu, “End-user programming of visualisations,” in *2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pp. 248–249, IEEE, 2016.